

VMC-7440

VMC-7440 – v8.0.6 Software Delivery Description and Instructions

List of Revisions			
Rev	Pages Revised	Description of the Revision	Date
1.0.9	All	Original	20/04/2009

Table of Contents

1.	Copyright Notice	3
2.	Introduction	3
3.	Product description	4
4.	Package Description	8
5.	Installation	9
5.1.	Hardware installation	9
5.2.	Software Installation	9
5.3.	Application Installation and Registration	15
6.	Livewire Sample Application.....	17
6.1.	Registry Settings	17
6.2.	Known Limitations	18
6.3.	Graphic User Interface Description.....	18
6.4.	Starting with the application.....	23
7.	Parameters Configuration	27
7.1.	Introduction to VMC-7440 SDK.....	27
7.2.	XML configuration file	27
7.3.	Assembly selection	27
7.4.	Capture Ch1 and Capture Ch2 selection.....	28
7.5.	IO selection	29
7.6.	Livewire Setting page.....	29
7.7.	List of parameters by component	31
7.8.	Frequently Asked Questions:	38

1. COPYRIGHT NOTICE

VITEC AND ITS AFFILIATES ("COMPANY") FURNISH THIS DOCUMENT "AS IS," WITHOUT WARRANTY OF ANY KIND. THE COMPANY DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT AND THOSE ARISING FROM A COURSE OF PERFORMANCE, A COURSE OF DEALING, OR TRADE USAGE. THE COMPANY SHALL NOT BE LIABLE FOR ANY ERROR, OMISSION, DEFECT, DEFICIENCY OR NONCONFORMITY IN THIS DOCUMENT AND DISCLAIMS ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHTS RELATED TO THE INFORMATION CONTAINED IN THIS DOCUMENT.

No license, expressed or implied, to any intellectual property rights is granted under this document. This document, as well as the software described in it, are furnished under a separate license and shall only be used or copied in accordance with the terms of the applicable license. The information in this document is furnished for informational use only, is subject to change without notice, and should not be construed as any commitment by the company. Except as permitted by the applicable license, no part of this document may be reproduced, stored in a retrieval system, or transmitted in any form or by any means without the express written consent of the company.

Names and logos identifying products of Vitec in this document are registered trademarks or trademarks of Vitec. All other trademarks mentioned in this document are the property of their respective owners.

Copyright © 2012 Vitec. All rights reserved.

Website:

<http://www.vitecmm.com/>

2. INTRODUCTION

This document details physical and software interfaces of VMC-7440 product and also offers a step by step guide package in order to describe how to install and use the product for basic usage.

It has also been written to enhance the user's experience of the product by introducing concepts of the Software Development Kit based on Livewire and settings configuration of the product

This beta version is not fully tested yet and it may have unknown issue

3. PRODUCT DESCRIPTION

VMC-7440 product is a high definition H.264 dual hardware encoding card based on VM4400 chip.

Designed for demanding professional video applications, this PCIe (1 lane) hardware-based encoder supports high profile with level up to 4.1 (i.e. HP@L4.1).

VMC-7440 SDK is based on Livewire technology which offers a way for software developers to add mpeg compression capabilities to their application from a set of ready to use software modules named Livewire components.

Livewire SDK allows a full and independent control of all the different features of the product (video compression/live video preview/still image capture and so one) through the notions of assembly, branch and preset systems.

Vitec also offers a Visual Studio Wizard that allows software developer to implement their own algorithm into a custom Livewire component.

VMC-7440 card works in 1, 4, 8 or 16 lanes PCI express connector.

VMC-7440 is composed on one IO board named VML-7040 and one compression module named CM-0410 board. Each board has its own sticker with name of the card and serial number which should be provided when requesting a RMA number to your reseller.

Dimensions of VMC-7440 are 180 mm(width) x 111.15mm (height)x 21.59mm (deep).

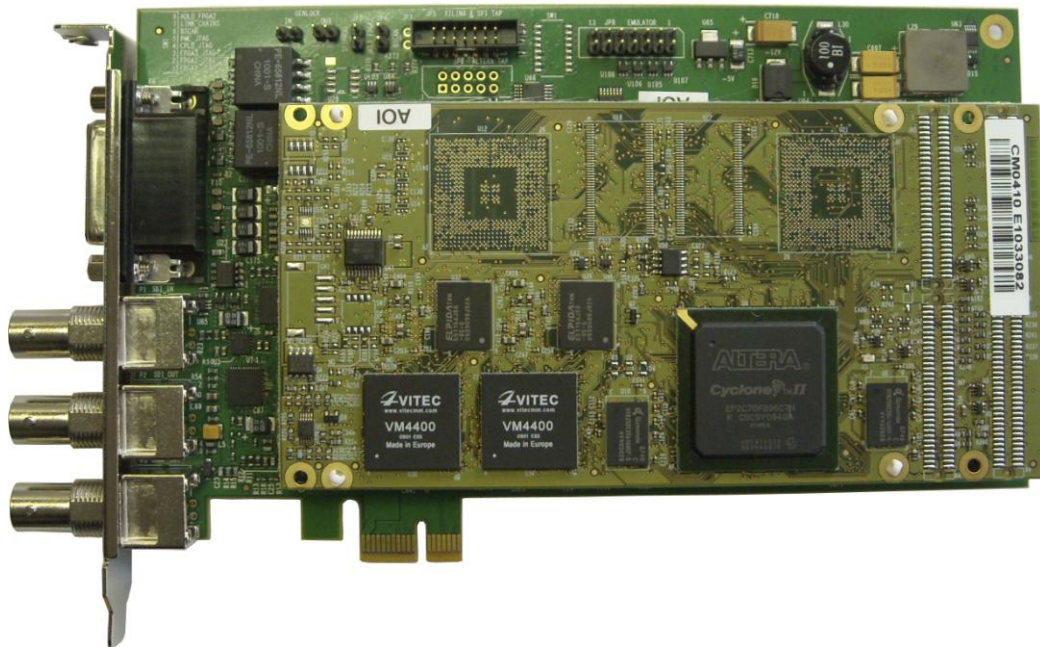
This product consumes up to 16W in dual encoding mode.

Features of the product:

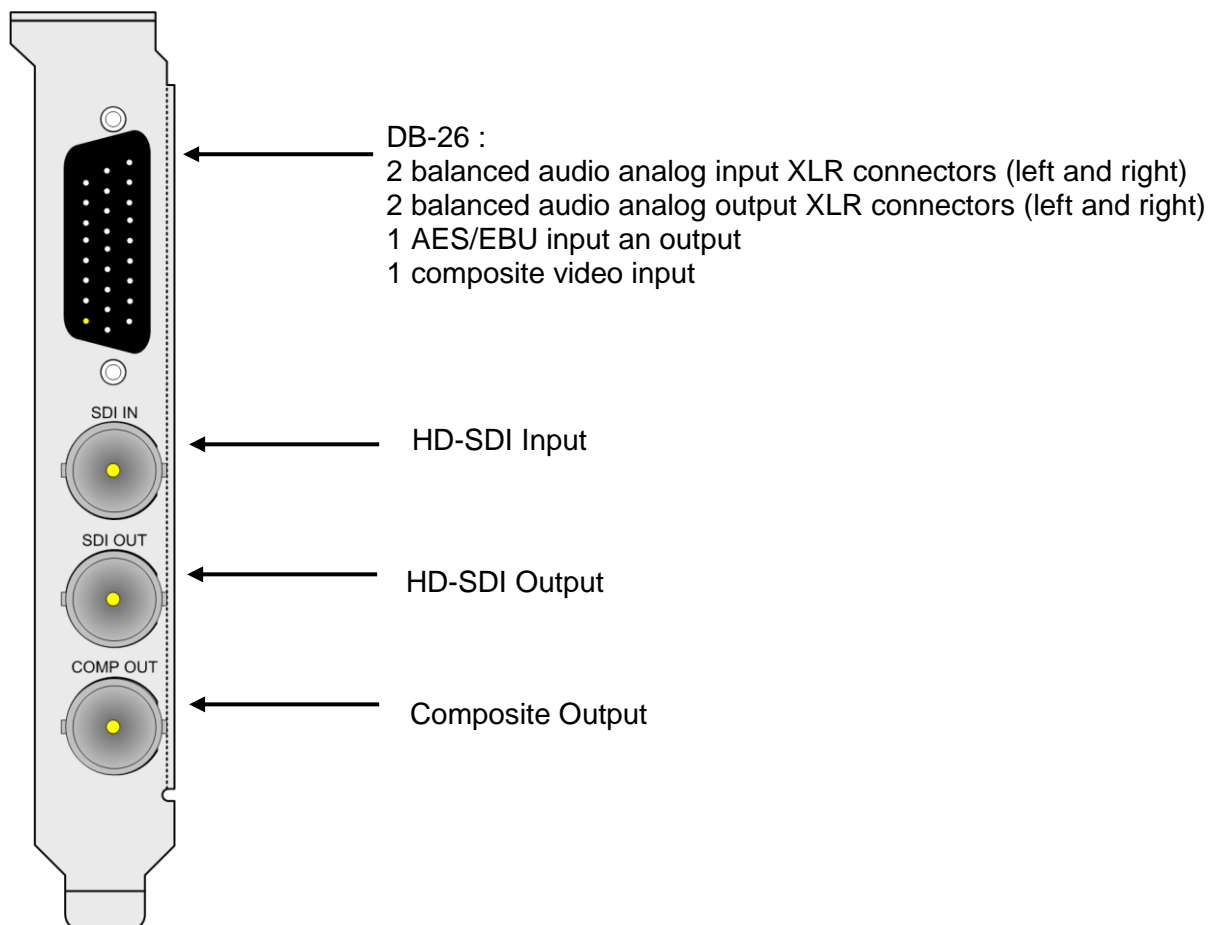
- PCI-e 1x. Interface
- HD-SDI Input supporting:
 - 1080p 23.976, 24, 25, 29.97, 30 formats
 - 1080i 23.976, 24, 25, 29.97, 30 formats
 - 720p 23.976, 24, 25, 29.97, 30, 50, 59.94, 60 formats
 - NTSC with 487 or 507 active lines.
 - PAL
- Composite Input on DB connector (see optional cable) with PAL and NTSC support.
- Only one video capture at a time (either SDI or Composite)
- HD-SDI and Composite Pass-through on video outputs
- Analog and AES/EBU Pass-through on audio outputs
- HD-SDI and Composite Auto Detection
- Analog, AES/EBU and Audio Embedded support(HD/SD-SDI) for audio inputs
- WinXP/Vista/7/Win 8/Server 2003 and 2008 driver support for 32 and 64 bits versions.
- Dual Video MPEG 4 AVC Encoder up to 1080p30, High Profile Level 4.1 with Pause/Resume/Split capabilities.

- Downscaling/Up scaling support for MPEG encoding (only interlaced -> interlaced and progressive -> progressive).
- Support constant or variable bitrate for MPEG encoding.
- Hardware Stereo Audio Encoder in AAC and MPEG1 Layer II formats.
- Capture still image in bmp format and encode MPEG files on hard drive
- MPEG 2 TS multiplexer
- MP4 container
- Video preview in native or downsampled to SD resolution
- Livewire SDK: independent control of different features in time, configuration through XML configuration file
- Audio level adjustment in dB
- Hue, saturation, brightness, contrast and gamma adjustments.
- Audio format detection (PCM, AC-3)
- Encoder continues to encode when input is unplugged while encoding
- Status monitor in video preview. (encoding time overlaid in preview)
- Overlay monitor in video preview. (audioMeter/VU overlaid in preview)
- OSD to overlay Unicode text or picture from 24 bits bmp or 32 bits tga TrueVision formats.
- Video preview in DirectDraw and Direct 3D mode with adjustable aspect ratio.
- C++ and C# Sample application with source code.
- Support of up to 16 VMC-7440 cards in the same system
- Vertical Interval Time Code captured and stored in Pict Timing message "ISO/IEC 14496-10 (ITU-T H.264), International Standard (2005), Advanced video coding for generic audiovisual services."
- Close Caption capture and store in SEI messages (refer to ITU-T Rec. T.35 SEI units).

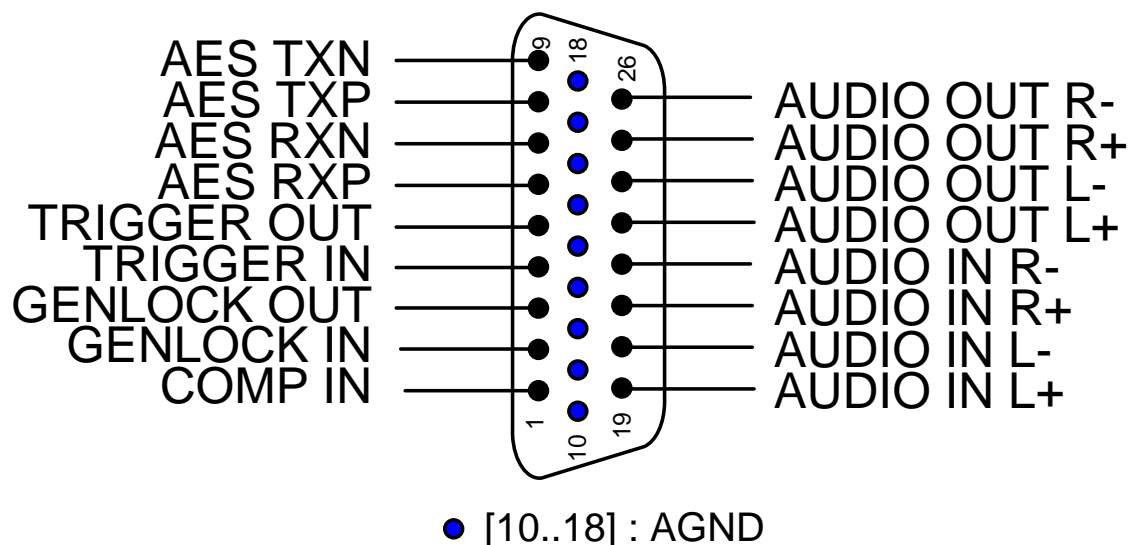
Overview of VMC-7440 card:



Bracket Description :

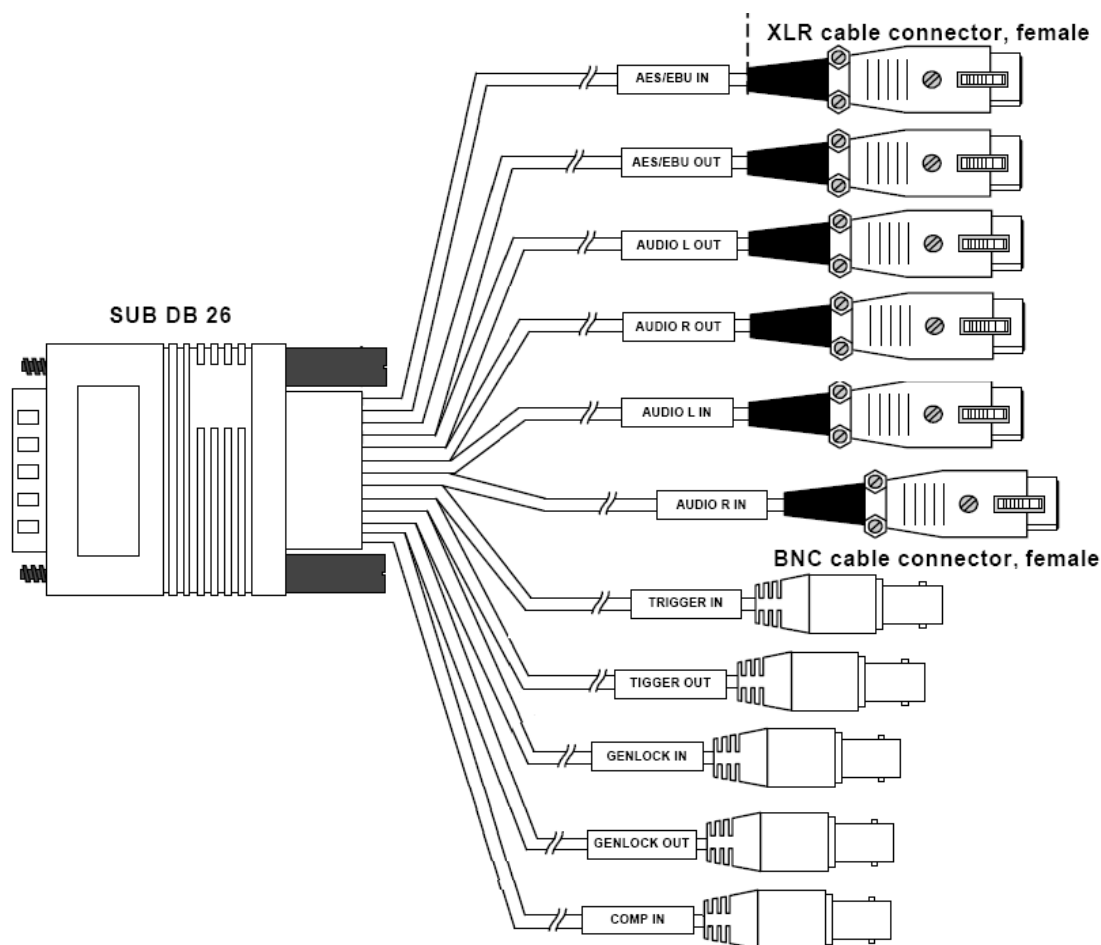


DB Description :



Optional Cable VML-7040:

IMPORTANT GENLOCK and Triggers inputs and outputs are not implemented.



WARRANTY:

VITEC guarantees its products against hardware or manufacturing defects for a period of twelve months from the date of purchase. This warranty does not cover accidental damages resulting from improper use of the product or from changes or repairs carried out by a non-approved facility.

4. PACKAGE DESCRIPTION

The package is delivered as a zip file named VMC-7440_SDK_v_8_0_3_Beta.zip.

This software delivery has been designed to work on Windows operating systems in 32 and 64 bits with Visual Studio 2008 SP1 runtimes component. Other operating systems are not supported.

This archive includes the following folders and files:

\Product Files\Bin

- Livewire Sample Application.exe : Main Sample Application written in C++ language
- C Sharp application.exe : Application written in C Sharp
- App.conf : Configuration file for C Sharp application.
- *.dll : Application libraries
- VMC7440 segmented unicode.xml: Configuration file for livewire test application.
- VMC7440 dictionary EN.xml. Configuration file that contains the display named in Livewire settings page.
- register.bat : batch file for DLL registration
- unregister.bat : batch file for DLL uninstallation
- mcmp4mux.dll: library, used for MPEG4Multiplexer.dll

\Product Files\VC 2008 redistrib

- **vc redistrib_x86.exe** - runtime components of Visual C++ Libraries required to run applications developed with VMC-7440 SDK on a computer that does not have Visual C++ 2008 SP1 installed. (NOTE: sp1 redistrib package is required. It includes 32 and 64 bits OS run-time)

\Product Files\Driver

- **Win7 contains** Windows 7 32 and 64 bit drivers.
- **WinServer2003 contains** Windows Server2003 32 and 64 bit drivers
- **WinServer2008 and Vista contains** Windows Server2008 and Vista 32 and 64 bit drivers
- **WinXP Vista contains** Windows XP 32 drivers **only**.

\Dev Resources\LivewireSDK\samples\C++ Sample Application

- Sample application built from C++ and its source code.

\Dev Resources\LivewireSDK\samples\C Sharp Sample Application

- Sample application built from C# and its source code.

\Dev Resources\LivewireSDK\ComponentWizardVC6

- Visual Studio 6.0 wizard for creating new Livewire component

\Dev Resources\LivewireSDK\ComponentWizardVC9

- Visual Studio 2007 wizard for creating new Livewire component

\Dev Resources\LiveWireSDK\LivewireSDK

- Lib, headers and samples to develop your own LiveWire component for VC6 and VC9 Environments (see Tutorial for explanations)
- Headers to compile sample application and your own application component for VC6 and VC9 Environments.

\Dev Resources\LivewireSDK\doc\Tutorial

- Tutorial describing the LiveWire architecture and how to use it.
Start with “LiveWire Tutorial.html” first!

\Dev Resources\LivewireSDK\samples

- Some examples of livewire client application and components.

5. INSTALLATION

5.1. Hardware installation

- 1 – Shutdown the computer from the operating system.
- 2 – Before any modification inside your system, make sure power is off. Remove the AC power cable once the computer has safely shut down.
- 3 – VMC-7440 card is ESD sensitive so pay attention while handling it. Electrostatic Discharge (ESD) can quickly and easily damage or destroy your card or your computer.
- 4 – Position the VMC-7440 above a PCI express connector and gently push down the card until the top of the card bracket is directly in contact with the chassis.
- 5 - Fasten the VMC-7440 card to the computer case with a screw in the chassis. Some new PC cases may provide a tool free system that holds the card in place without any screw.
- 6 – Plug your HD-SDI and audio cables to the inputs and outputs of the VMC-7440.

5.2. Software Installation

Extract the archive to a working directory of your choice except <system32> folder since it may causes technical issues during software updates.

If the VMC-7440 board is installed for the first time, Windows will pop up the Plug And Play installation wizard. In such a case please proceed like this:

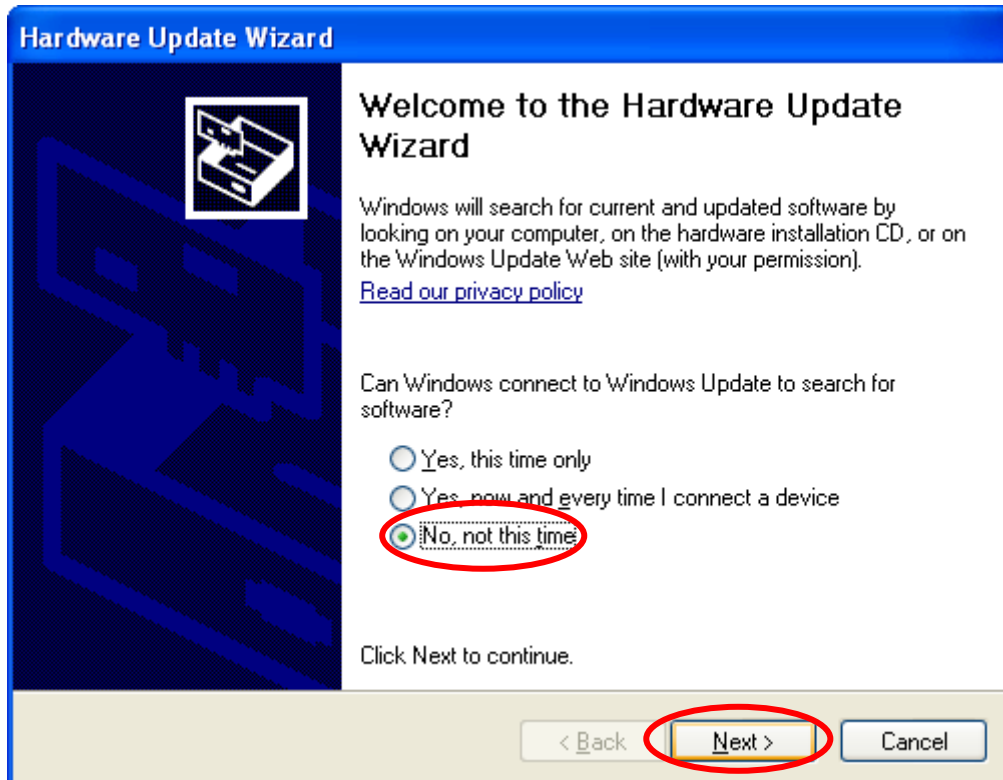
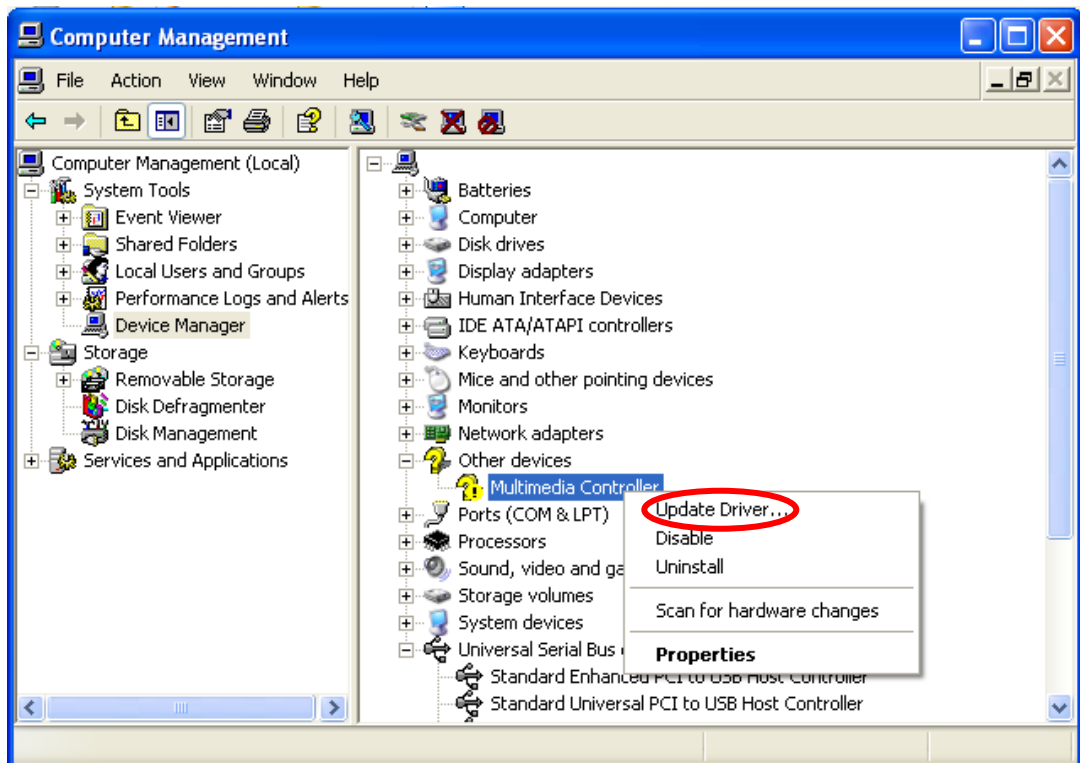
Extract the archive to a working directory of your choice.

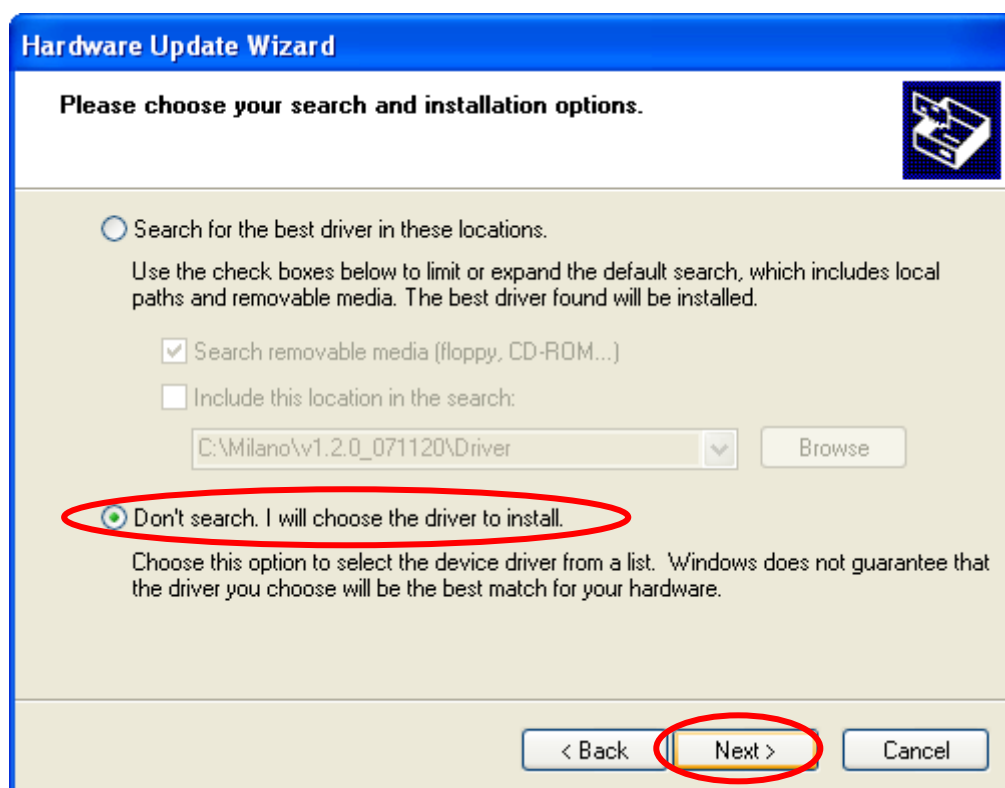
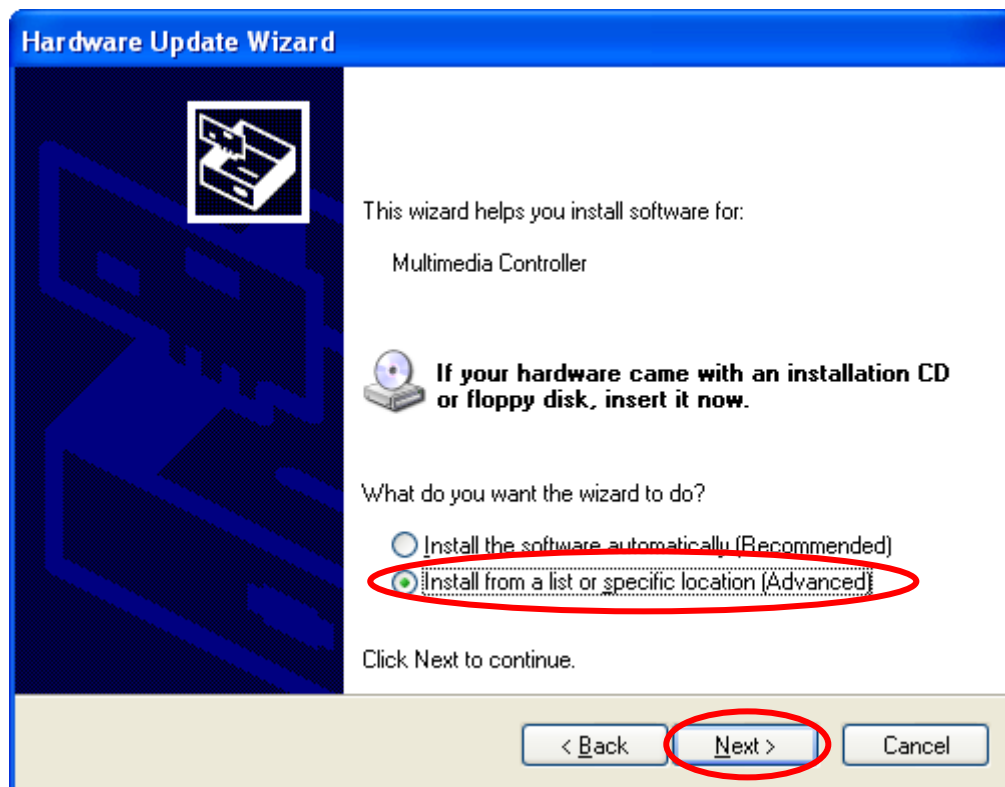
From the wizard window, browse to the “<working directory>\Product Files\Driver\” folder, point to the “inf” file in the directory named following your operating system version and proceed with the driver installation.

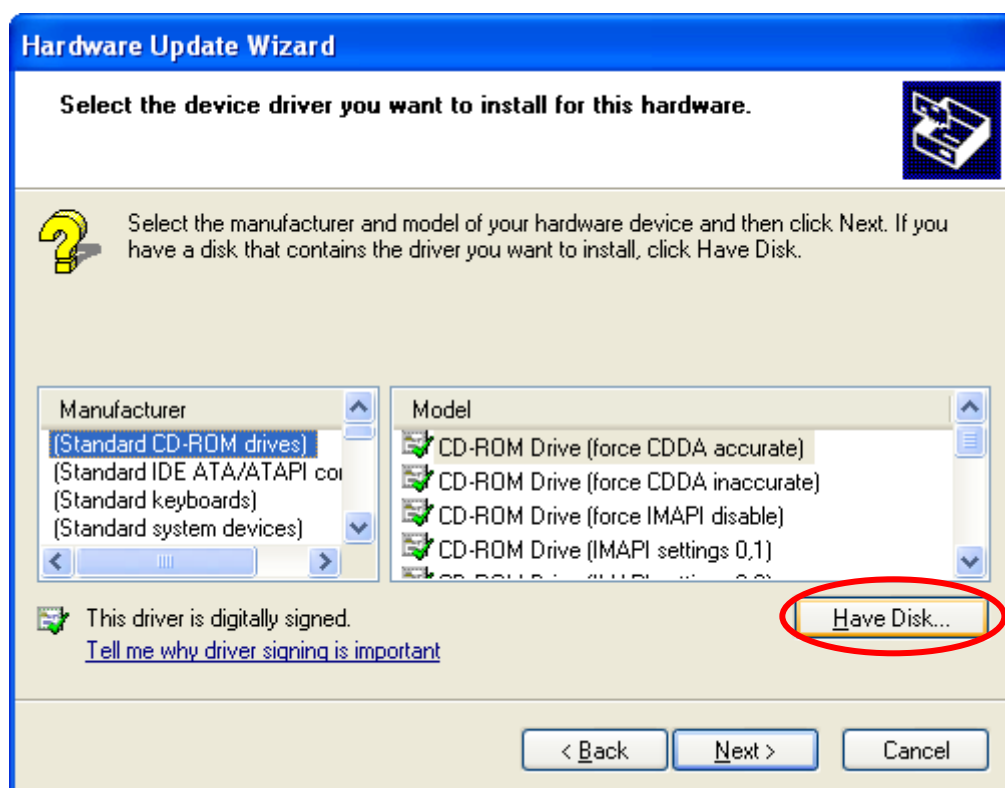
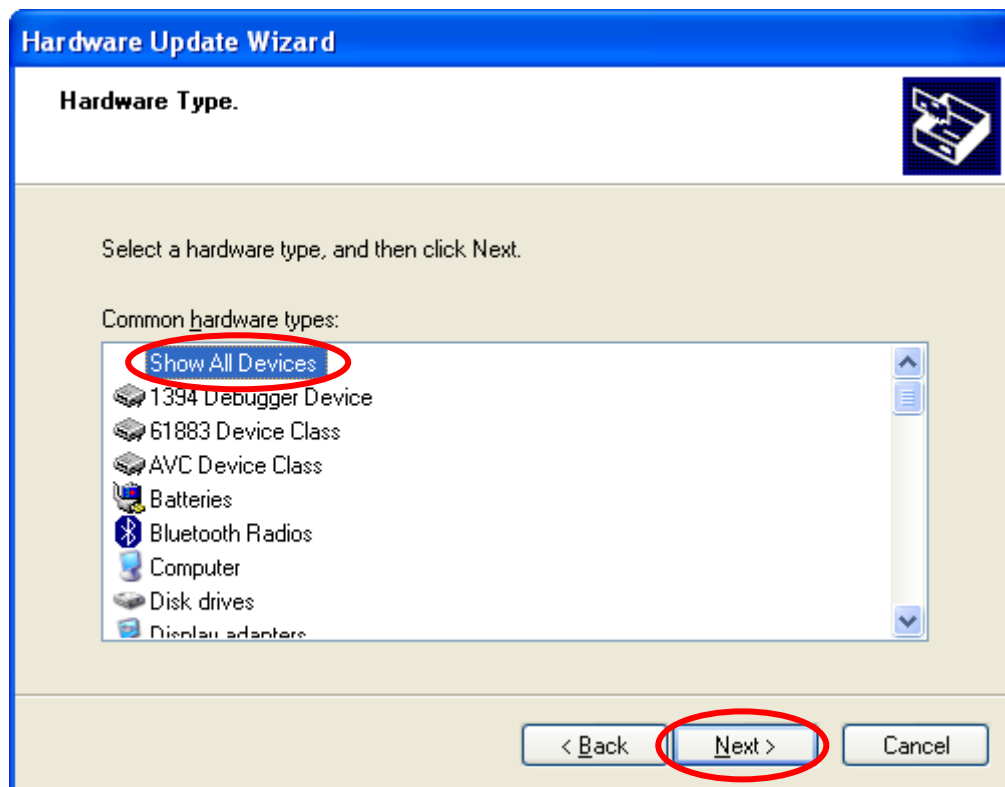
If the VMC-7440 board has already been installed with a previous software package, then run Windows Device Manager to manually update the driver and point to the “<working directory>\Product Files\ Driver\” folder.

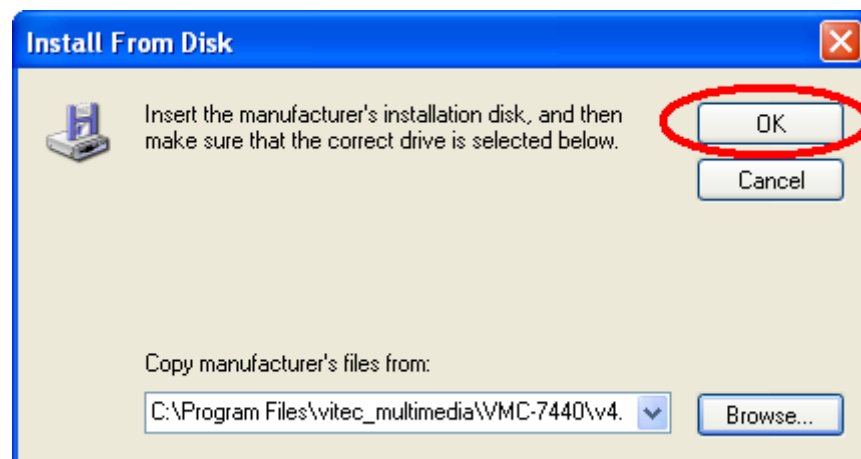
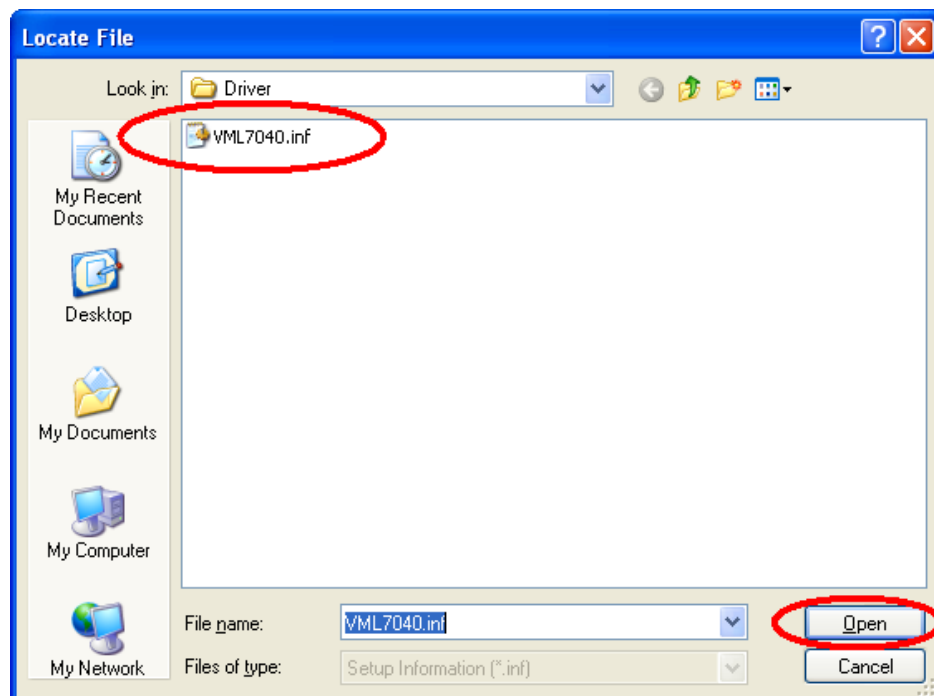
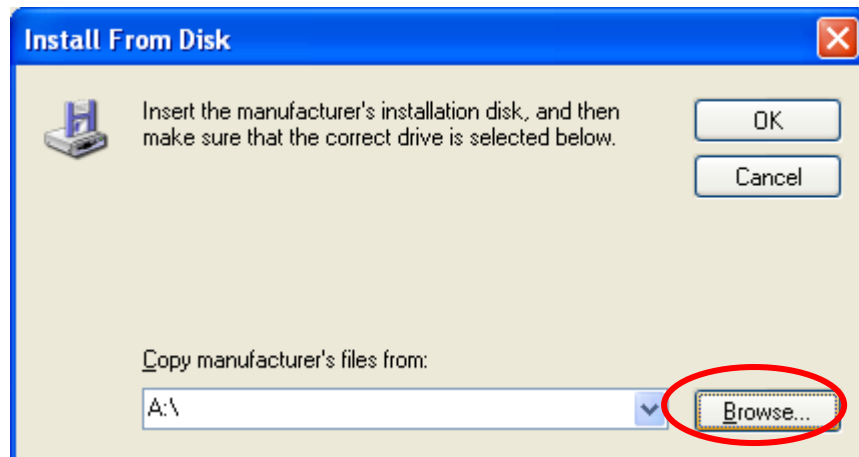
Please follow the steps below:

Note: The product version mentioned on the following screenshots might not be up-to-date. However, the step-by-step description still applies so please follow this procedure.

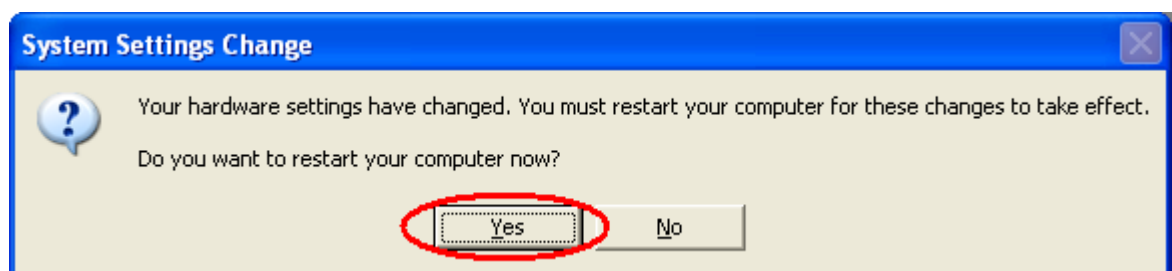
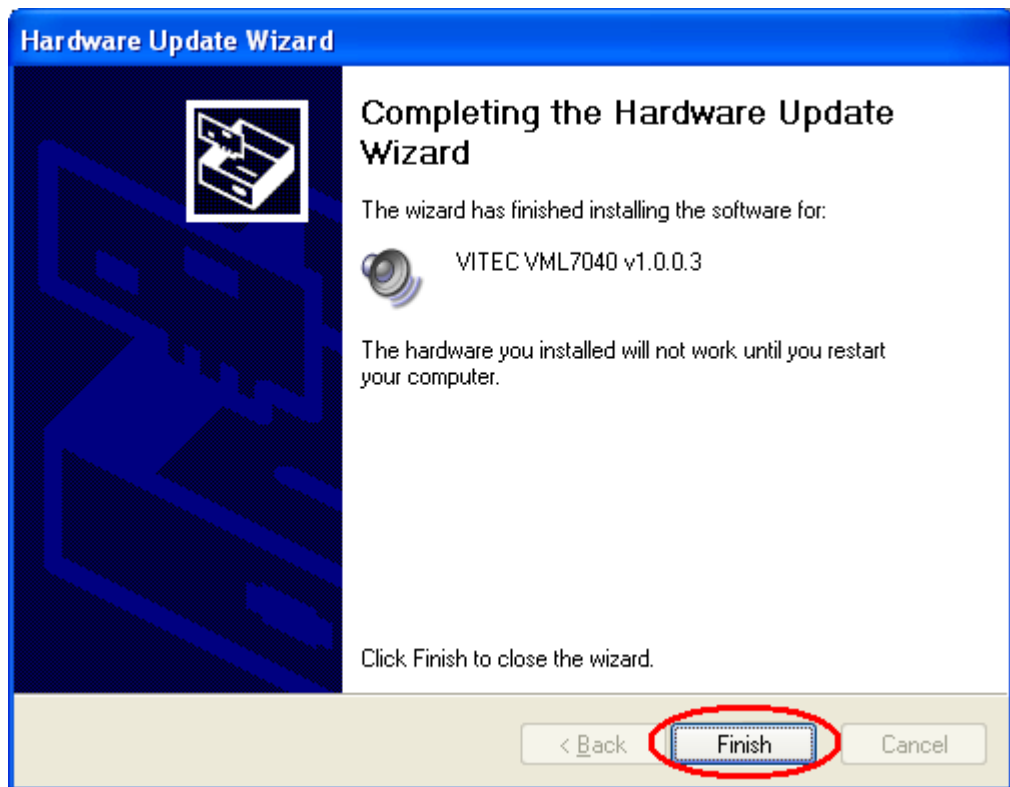








Depending on installed operating system, you might have a Security warning that tells “Windows can't verify the publisher of this driver software”. In this case, please confirm installation of the driver to continue.



5.3. Application Installation and Registration

IMPORTANT NOTE about Side by Side installation:

The ComponentDependency.manifest file delivered in the Product_Files\Bin folder replaces the registration process through the register.bat file.

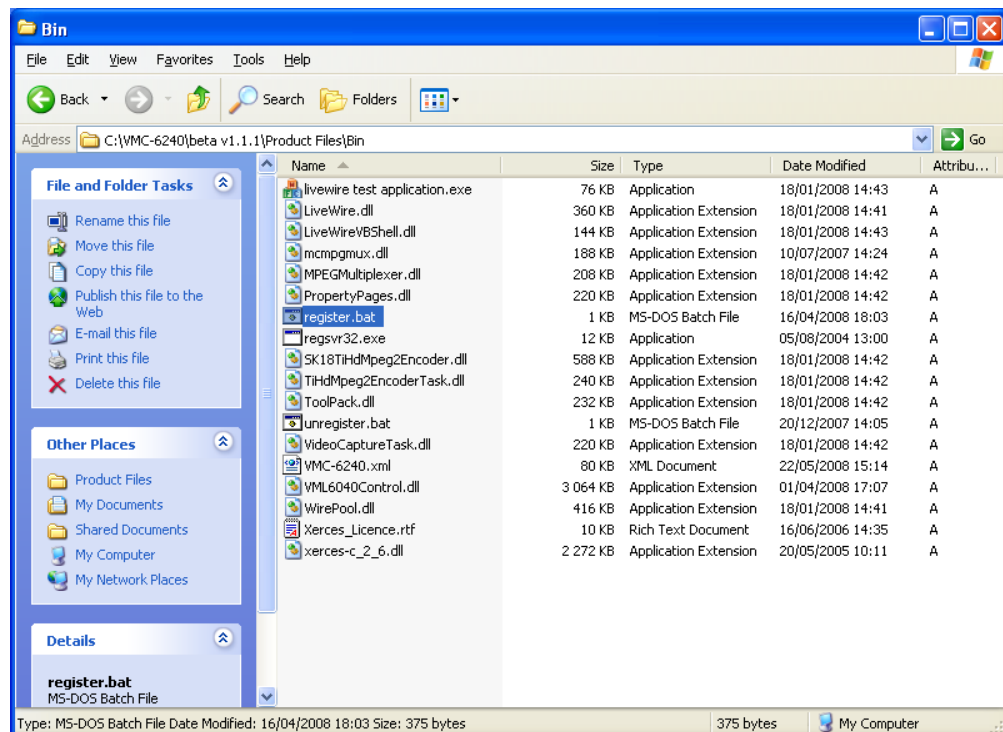
Please note that the manifest file is not compatible with the Vitec C# shell.

Please follow the registration process as described below only if you:

- if you want to use the C Sharp Application.exe.
- if you develop a new application I C#.
- if you use an application that does not refer to any manifest

Note: The sample application written in C++ delivered with this SDK (livewire sample application.exe) refers to ComponentDependency.manifest file. It means that registration of dlls is not mandatory before starting this application.

Run "<working directory>\Product Files\Bin\register.bat" to register DLLs.



Please check that each message returns a successful registration.

If a dll registration failed with an error/unsuccessful message, please follow the steps below before registering these dlls again:

1. Verify dependencies of concerned dll.
2. Check the rights of the current User Account (administrator rights are required to properly install the board).
3. Turn off User Account Control (UAC) if you are using Microsoft Vista operating system. If you use Windows 7, please move the Slider of UAC to Never Notify state.
4. Modify register.bat to request Admin access before regsvr32 calls (see faq).

To turn off UAC:

1. Click Start, and then click Control Panel.
2. In Control Panel, click User Accounts.
3. In the User Accounts window, click User Accounts.
4. If you use windows 7, click on the User Account Settings link
5. In the User Accounts tasks window, click Turn User Account Control on to off for Vista. If you use Windows 7, please move the Slider of UAC to Never Notify state.
6. If UAC is currently configured in Admin Approval Mode, the User Account Control message appears. Click Continue.
7. Clear the Use User Account Control (UAC) to help protect your computer check box, and then click OK.
8. Reboot the system when prompt and run register.bat again.

Now the product is correctly installed.

To unregister the Application dlls, you can use the unregister.bat in the same folder.

6. LIVEWIRE SAMPLE APPLICATION

The Livewire sample application is a demo application written in C++ language which is common to several other Vitec products such as VMC-7440, VMC-6240 and VM family.

The goal of this application is to be the reference in term of syntax for software developers who would like to integrate VITEC product in final application with a custom graphic unit interface

Therefore, this application might not fulfil requirement of final customer in terms of graphic interface or advanced features. In the same time you might find pieces of code which are not used in the context of this particular product.

Livewire sample application loads an XML configuration file to construct and control card functionality. Including: Still Image capture in bmp format, MPEG encoding, Video Preview rendering. LiveWire Setting page is accessible for product parameters control.

The Visual Studio 2008 project with all source code of the application is available in \DevResources\Cpp Sample Application subfolder.

6.1. Registry Settings

Livewire Sample Application stores its application parameters in HKEY_CurrentUser/Software/Vitec Multimedia/livewire sample application path of Windows registry:

Capture Duration: (see Item 23 of paragraph 6.3)

Capture Segments: Not Implemented

Grab File Name: (see Item 8 of paragraph 6.3)

Mpeg File Name: (see Item 9 of paragraph 6.3)

Properties File Name: default path for loading a XML file at startup.

Still Image Distance: Not Implemented

Still Image Number: Not Implemented

Windows X: Horizontal position of graphic user interface.

Windows Y: Horizontal position of graphic user interface.

Please remove all Keys from “HKEY_CurrentUser/Software/Vitec Multimedia/livewire sample application” if this application does not appear at startup.

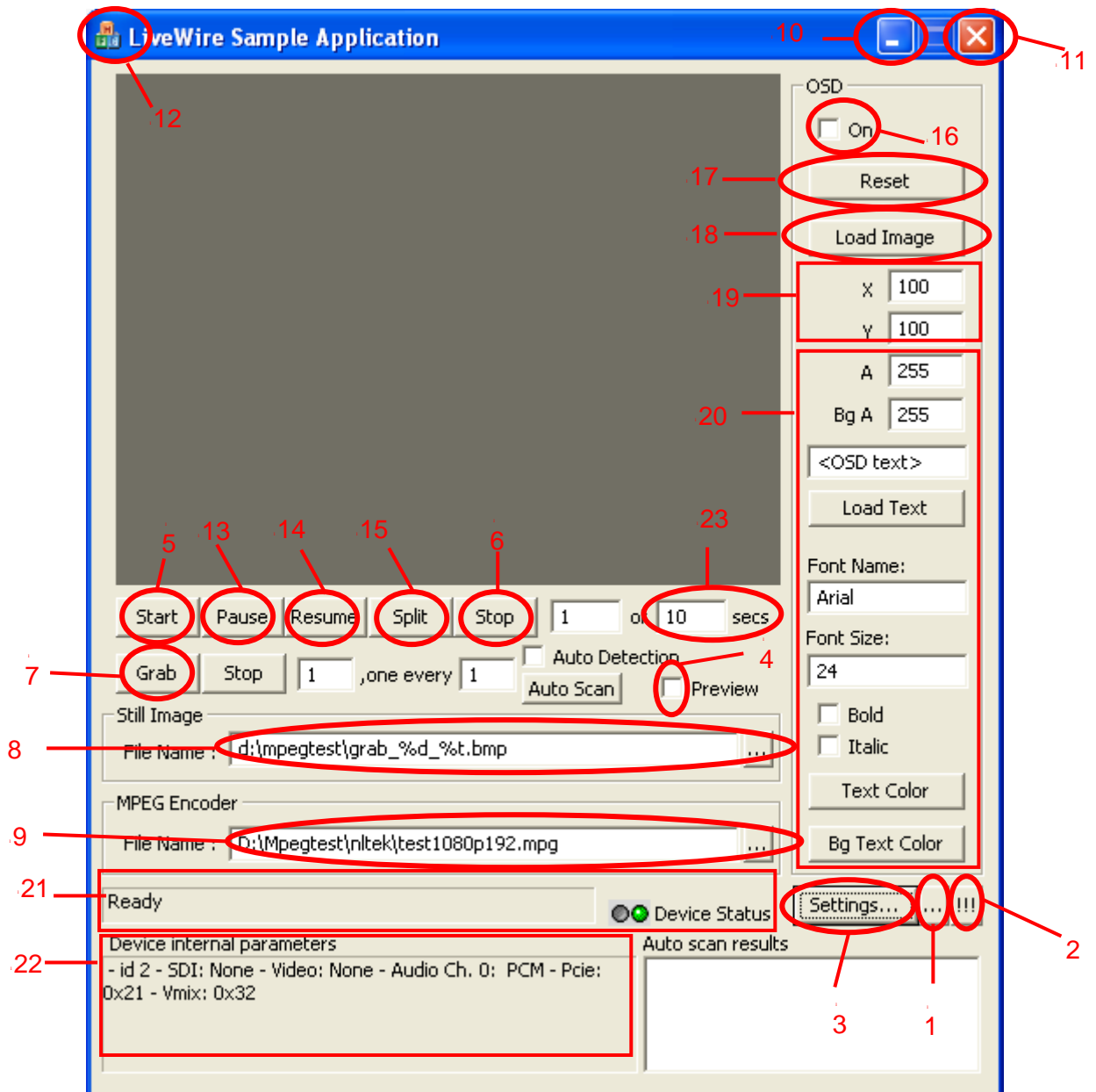
6.2. Known Limitations

Please refer to the readme for the list of known issues and limitations.

6.3. Graphic User Interface Description

Run “<working directory>\Product Files\Bin\Livewire Sample Application.exe” to start the application. After few seconds, the Graphic User Interface should pop up on the windows desktop.

The following screenshot shows operational controls circled in red:



- 1 – Browse and load one XML file.
- 2 – Save current settings as a new XML file.
- 3 – Open Livewire settings page that allows configuring the product.
- 4 – Start or stop the “Preview” branch to control the video preview window.
- 5 – Start the “Capture” branch to manually start MPEG encoding.
- 6 – Stop the “Capture” branch to manually stop MPEG encoding.
- 7 – Start the “Grab” branch to capture a still image in bmp format.
- 8 – Set path and file names for still image captures.
%d, %t and %0 can be used as wildcards for date, time and index number.
- 9 – Set path and file names for MPEG files.
%d, %t and %0 can be used as wildcards for date, time and index number.

Please note that pathname for bmp and mpeg files must exist before capturing image or recording video file. Output directory is not created by the application.

- 10 – Minimize the GUI
- 11 – Close Application
- 12 – Open Additional menu.
- 13 – Pause the MPEG encoding.
It allows skipping undesired video until resume command is executed.
Pause and Resume commands requires presence of StreamRegulator component in current assembly.
- 14 – Resume for a pause.
- 15 – Split is not operational with VMC-7440.
However, it is possible to use a feature named auto split if StreamRegulator component is present.
Auto split automatically stops the current encoding after a specified duration (defined in “Cut each X Seconds” parameter of StreamRegulator) and start a new file without any loss of video frames or audio.
- 16 – OSD checkbox. This control allows starting or stopping the OSD branch. It is mandatory to check this button in order to send OSD command.
- 17 – Reset OSD. This control allows erasing all the current OSD content.
- 18 – Load a picture in OSD memory. Bmp can be in 24 bits bmp format or tga format. Tga format includes transparency effect. Picture horizontal and vertical dimension must be a multiple of 4 pixels. Please remember that loading a picture in OSD consumes CPU.

19 – OSD position. Set horizontal and vertical position for the next OSD object (text or picture). The starting point is the left top corner of the captured video.

20 – OSD Text controls.

“A” field allows setting text and picture transparency on 16 levels (from 0 to 255). 255 value is fully opaque. “Bg A” field allows setting text background transparency on 16 levels (from 0 to 255). 0 value means that you will see only text character on video without any background.

Other buttons allows changing font, size, colors and style (italic or bold)

Replace **<OSD Text>** by the text string to overlaid.

Load Text button will apply the text following the current configuration.

21 – Display “Device Status” parameter status. This parameter of the device component indicates the current state of the communication between card and system. It can take following values:

- 0 for "OFF"
- 1 for "Ready"
- 2 for "In Progress"
- 3 is not used
- 4 for critical error.

Application should verify that this parameter is in ready state before applying any command.

In the same way, application should wait for “Device and Encoder ready” event (code 0x12000000) before applying any command to the encoder.

22 – Device Internals parameters.

This section display values of some basic status parameters of Device component such as:

“VideoMixer Microcode Version” – This parameter indicates version of bitstream of video mixer module. This bitstream is loaded by application.

“SDI Camera” – refer to note 1&2 hereunder.

“Analog Camera” refer to note 1&2 hereunder.

“AudioFormat1” – This parameter indicates type of embedded audio. (PCM or AC3)

“Pci Microcode Version” - This parameter indicates version of bitstream of IO card (VML-7040 card). This bitstream is stored on a flash of the IO card.

“Board Id” – Give hardware revision of IO card.

These parameters are not visible in the Livewire Setting pages as they are declared as Internal. However, application can retrieve their value through get_ComponentParameter method from Assembly Container.

23 – Capture Duration.

This parameter set duration in second of next MPEG Encoding. 0 means that encoding will not stop automatically.

Note1: The board device has 2 parameters named “SDI Camera” for SDI detection value and “Analog Camera” for composite detection value.

The SDI video resolution detection value could be:

- *DIN_1080I (2048): 1920x1080, 2 fields*
- *DIN_1080P (1280): 1920x1080, 1 frame*
- *DIN_720P (768): 1280x720, 1 frame*
- *DIN_576I (2304): 720x576, 2 fields*
- *DIN_480I (2560): 720x487, 2 fields*
720x507, 2 fields

The composite video resolution detection value could be:

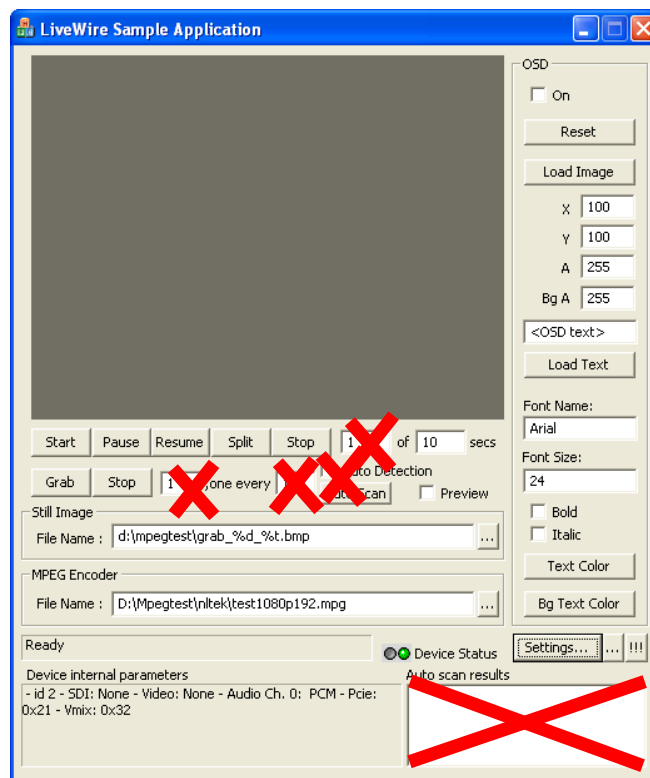
- `DIN_NTSC_NOCOLOR` (518): 720x487, 2 fields
- `DIN_NTSC` (516): 720x487, 2 fields
- `DIN_PAL_NOCOLOR` (517): 720x576, 2 fields
- `DIN_PAL` (515): 720x576, 2 fields

If the signal format is undetermined, detection routine returns the value `DIN_UNKNOWN` (1).
If there is no signal detected the detection routine returns the value `DIN_NONE` (0).

Note2: The assembly container object asks the device component to get value of detected input formats. When the parameter value changes, the device component sends a notification. The application could retrieve this event, and get the detection value store into the last 24 bits of `hrHresult` parameter from `VIT_EVENT_LOG_ENTRY` structure.

Note3: A hardware frame buffer upstream to the hardware encoder keeps the last good captured frame when video source becomes not operational for any reason. Encoder keeps encoding the last good captured frame even when there is no video source. An event signal warns application or user that there is no detected signal in input.

The next GUI snapshot shows unavailable features as marked with a red cross:



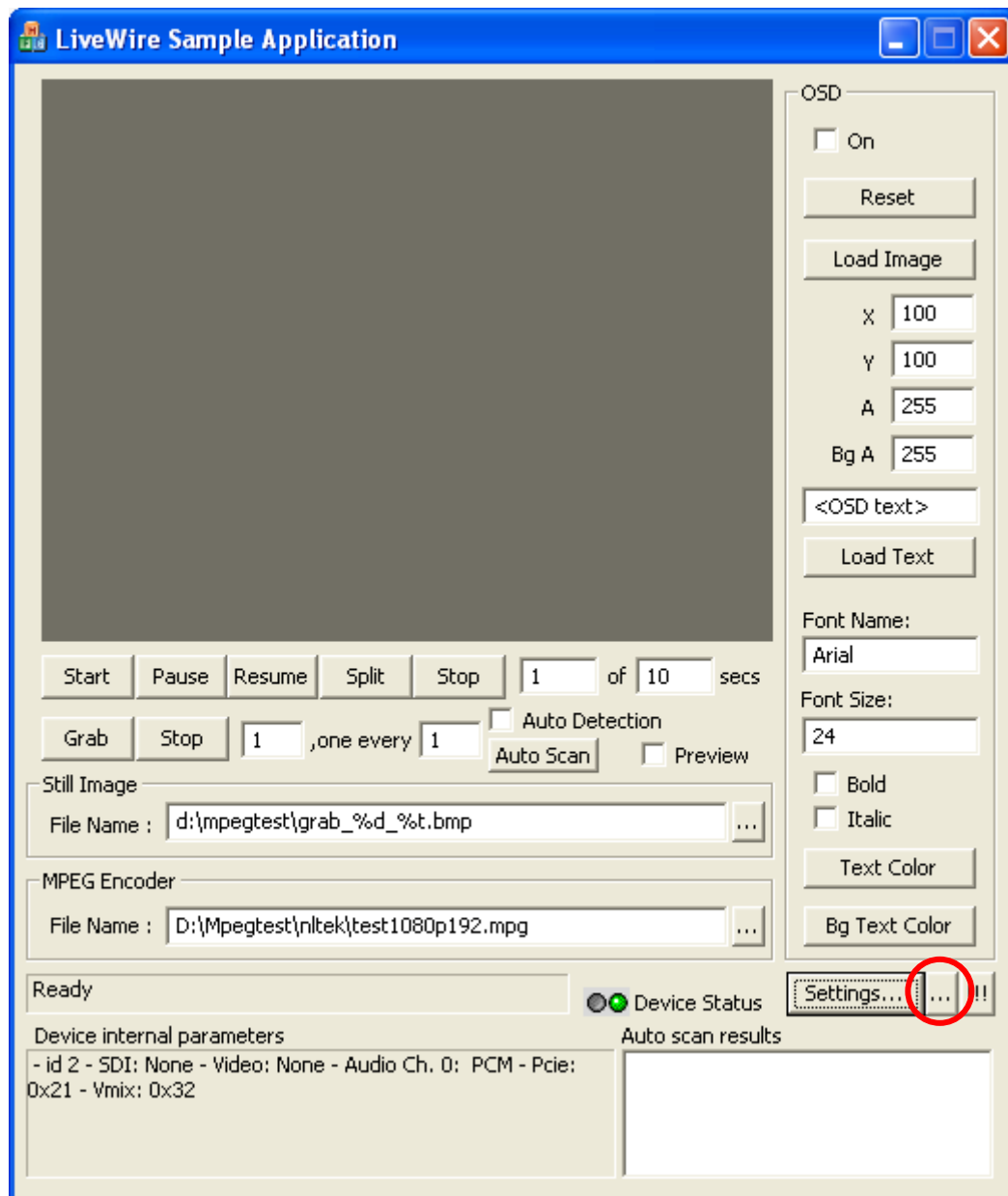
Multiple Grab – These two fields next to the “Grab” button are not available. They are designed to schedule multiple still image captures (with a defined interval between each). **These 2 parameters have to be set at “1, one every 0”.** Starting with the application

6.4. Starting with the application

6.4.1. Start the sample application

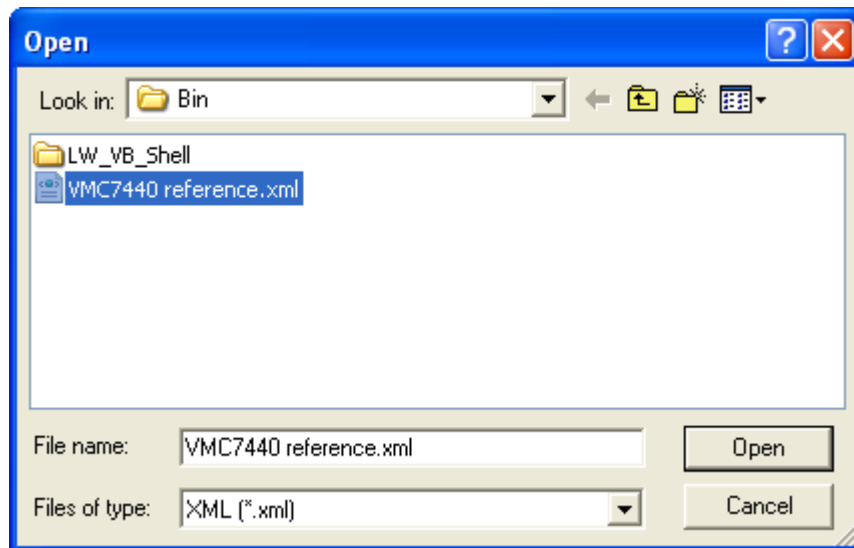
Run “<working directory>\Product Files\Bin\Livewire Sample Application.exe”

You should see the following screen:



6.4.2. Load the reference XML configuration file.

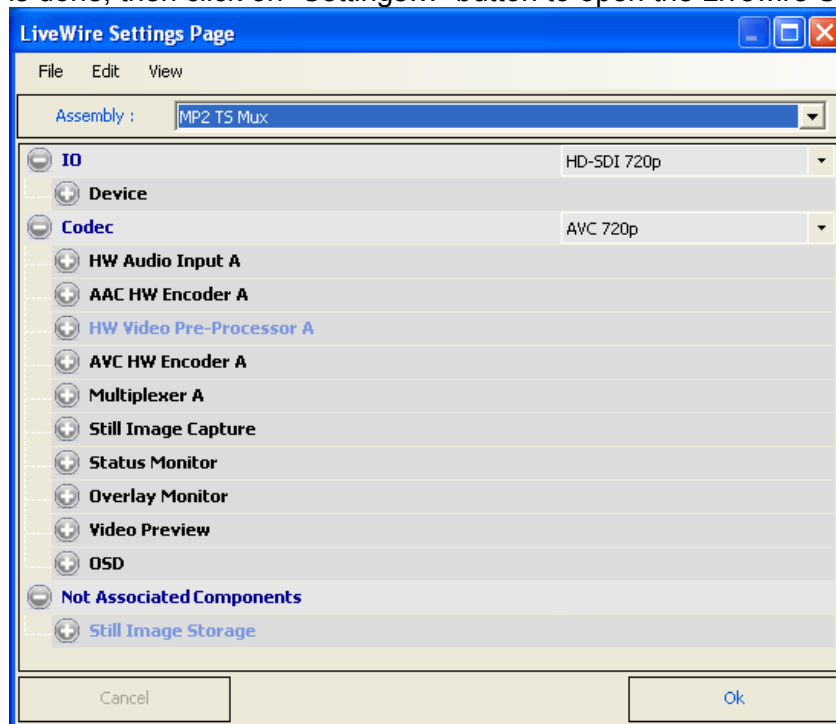
For the first use OR if a previous version has already been installed and executed on the computer, click on the (...) button to open a dialog box (make sure Preview check box is not selected), and browse to the "<working directory>\Product Files\Bin" directory to select "VMC-7440.xml reference" file:



Click "Open" to load the XML configuration file and return to main screen.

6.4.3. Open the LiveWire Setting Page

Once this is done, then click on "Settings..." button to open the Livewire Setting page



As described previously, the Livewire Setting Page allows user to configure the software configuration of the product.

6.4.4. Select an assembly.

User can select the assembly to work with.

Assembly defines the operation mode of the product. (See paragraph 7.)

6.4.5. Configure the video and audio input format.

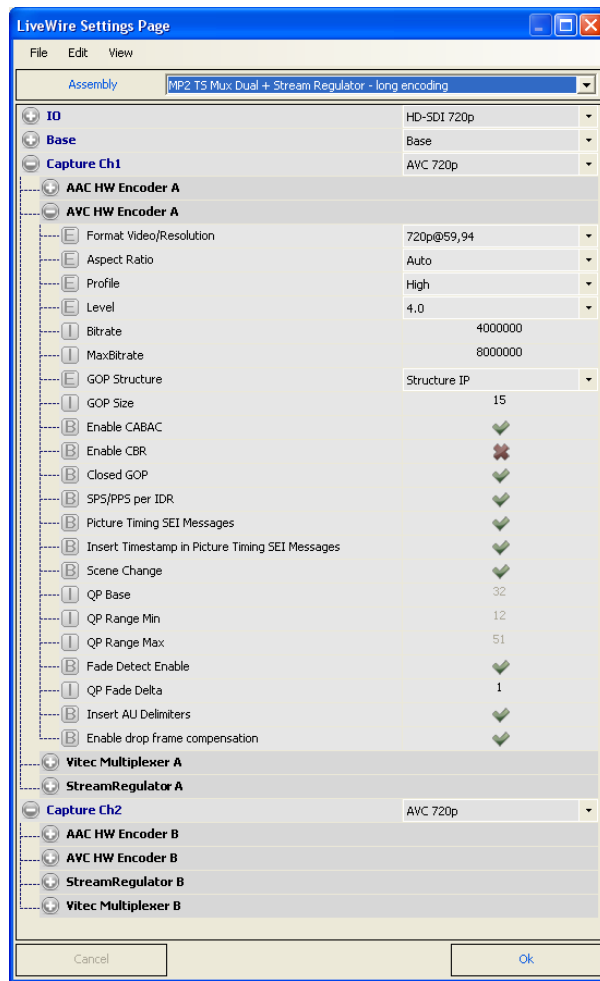
Just below, user can select the “**IO**” preset. To capture properly a video signal with this application, user has to select the “**IO**” preset following the detected format shown in the **Device Internals parameter**.

If you click on the “**IO**” node, you display the device section. . You can here expand device section to list all parameters show in a list box and then you can select the relevant video input following detected format.

If necessary, it is possible to fine tune input configuration by modifying Device parameter value.

6.4.6. Configure the “Capture Ch1” and “Capture Ch2” profiles.

Before encoding, user has to select the “Capture Ch1” and “Capture Ch2” presets below the current “**Base**” preset.



If necessary, it is possible to fine tune encoding settings by expanding components parameter in these sections of the LiveWire settings page.

This way, you can use downscaling capabilities of VM4400 encoder or modifying output bitrate.

Once it is done, user can exit LiveWire Settings page and use the different features available in the graphic user interface

For instance, click on the Preview checkbox to activate live video preview.

To encode some mpeg files, click (...) as shown below to select a file name and a destination folder and then click on the “Start” Button.

6.4.7. Play with the Graphic user interface.

Please refer to paragraph 6.3 Graphic Unit Interface Description for description of available features.

7. PARAMETERS CONFIGURATION

7.1. Introduction to VMC-7440 SDK

The VMC-7440 SDK uses the Vitec framework named Livewire which provides a set of ready to use software modules (also named Livewire components). Each Livewire component has been developed to perform a specific task of the product.

For instance, the “Video Encoder” component is responsible of the video encoding process whereas the “Storage” component writes data on hard disk.

All the different features such as MPEG encoding, live video preview, grab of still image or OSD control are structured into different branches such as “Capture” for the MPEG recording, “Preview” for the live video, “Grab” for capturing still image in bmp format or “OSD” for the OSD control. Therefore, a branch is a chain of components link together in order to perform a complex feature.

The different modes of the product (for instance changing the MPEG standard of the encoded file) are structured into assemblies which is a group of branches.

Application controls the different features through start, pause, resume or stop commands send to branches. This principle allows an independent control of the different features.

Each Livewire component can be configured through their own parameters. All parameters of a component are stored into a parameters section visible through the Livewire settings page under the parameter name. A segment gathers component parameters’ values in order to keep current values.

7.2. XML configuration file

XML files are configuration files used by each application developed with the VMC-7440 SDK. It describes available components, branches, segments and it also stores parameters within presets.

XML file can be adapted to a specific application. Sample application written in C++ is designed to work with the reference xml file (VMC-7440 reference.xml) delivered in this package. After initialization, XML content is interpreted and loaded into application memory and parameters can be modified by final user through the LiveWire Setting pages.

7.3. Assembly selection

An assembly is a set of components instantiated together for a specific usage. It is organized into branches allowing independent control of several features such as Video Encoding, Video Preview or Still Image grabbing by starting or stopping them individually.

Following branches are available for each assembly of VMC-7440 reference.xml :

- “Capture” branch controlling MPEG encoding.
- “Grab” branch controlling still image grabbing.
- “Preview” branch controlling Video Preview.
- “OSD” branch controlling OSD feature.

Six different assemblies are available in the “VMC-7440 reference.xml”:

- **MP2TS Mux**

This assembly allows encoding Transport Streams containing MPEG 4 AVC Video stream with AAC-LC Audio Streams encoded from Analog, AES/EBU and raw audio embedded in SDI (24bits@48kHz 2 channels).

- **MP2TS Mux + MPEG1 Layer 2**

This assembly allows encoding Transport Streams containing MPEG 4 AVC Video stream with MPEG1 Layer 2 Audio Streams encoded from Analog, AES/EBU and raw audio embedded in SDI (24bits@48kHz 2 channels).

- **MP4**

This assembly allows encoding files in MPEG-4 Part 14 format containing MPEG 4 AVC Video stream with AAC-LC Audio Streams.

- **A+V Streams**

This assembly allows encoding 2 different files for each encoding: one AVC encoded video file and one AAC encoded audio file.

- **MP2TS Mux Dual**

This assembly allows encoding 2 Transport Streams containing MPEG 4 AVC Video stream with AAC-LC Audio Streams encoded from Analog, AES/EBU and raw audio embedded in SDI (24bits@48kHz 2 channels). Both set of Transport Streams are encoded from the same source but they can be encoded in different format or encoding presets.

- **MP2TS Mux Dual+ MPEG1 Layer 2**

This assembly allows encoding Transport Streams containing MPEG 4 AVC Video stream with MPEG1 Layer 2 AAC-LC Audio Streams encoded from Analog, AES/EBU and raw audio embedded in SDI (24bits@48kHz 2 channels).

- **MP4 Dual**

This assembly allows encoding 2 files in MPEG-4 Part 14 format containing MPEG 4 AVC Video stream with AAC-LC Audio Streams. Both files are encoded from the same source but they can be encoded in different format or encoding presets.

- **A+V Streams Dual**

Same as A+V stream except that there is 2 Video Channel.

After selecting an assembly, the next step is to select a Codec preset.

7.4. Capture Ch1 and Capture Ch2 selection

Codec preset stores video encoders parameters.

Available Profile presets are:

- **AVC 720p** – Producing MPEG AVC Video when capturing 720p input format.
- **AVC 1080i** – Producing MPEG AVC Video when capturing 1080i input format.
- **AVC 1080p** – Producing MPEG AVC Video when capturing 1080p input format.
- **AVC PAL** – Producing MPEG AVC Video when capturing PAL input format.
- **AVC NTSC** – Producing MPEG AVC Video when capturing NTSC input format.

7.5. IO selection

IO presets store video input and audio input parameters.

Available Input presets are:

HD-SDI 720p – preset to capture 720p signals from HD-SDI input.

HD-SDI 1080i – preset to capture 1080i signals from HD-SDI input.

HD-SDI 1080p – preset to capture 1080i signals from HD-SDI input.

SD-SDI PAL – preset profile to capture PAL signals from HD-SDI input.

SD-SDI NTSC(487 lines) – profile to capture NTSC with 487 active lines signals from HD-SDI input.

This is the most popular NTSC format; please use it if you do not have any video capture issue with a NTSC source.

SD-SDI NTSC(507 lines) – preset to capture NTSC with 507 signals from HD-SDI input.

This mode allows backward compability with SDI NTSC devices using old interval field definition such as old betacam.

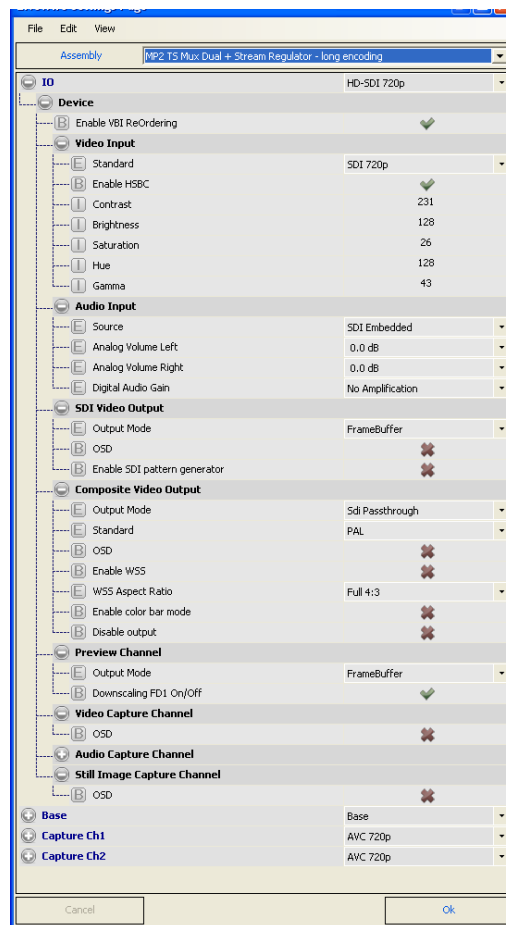
Composite PAL – preset to capture PAL signals from composite input.

Composite NTSC – preset to capture NTSC signals from composite input.

7.6. Livewire Setting page

The LiveWire Setting page configuration allows user to select Assembly, profiles and adjust values of parameters by component.

[Overview of the Livewire Settings Page:](#)



This property page is divided into 3 different sections:

- Assembly preset selection.

The user can change the mode of the product through this section.

- IO preset selection.

The user selects the desired preset of video input settings through this section

- Base preset selection.

The user selects the settings of Overlay, Status Monitor, Video Preview components.

- Capture Ch1 and Capture Ch2 preset selection.

The user selects the desired preset of video encoding settings for both video channels through this section.

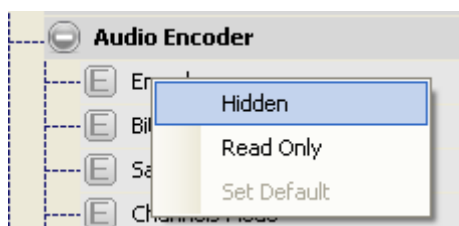
Parameters tree.

This part of the property page allows reading or changing a parameter value after expanding list of parameter of a component section.

A parameter value set its component behaviour or giving some status information for the user.

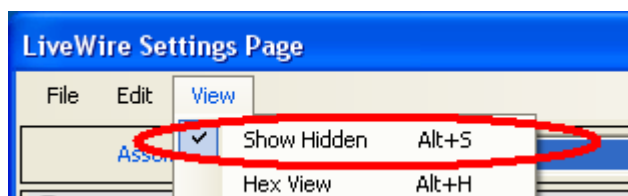
Parameters can have attributes such as Read Only and Hidden.

Attributes can be modified by clicking right on parameter in the Livewire Setting Page:



Parameters can be reset to default value through the Set Default button.

To show Hidden parameters, it is necessary to check Show Hidden in the View menu of LiveWire Setting page.



=> Please note that Livewire settings page cannot detect a bad configuration that would result in a malfunction of the product. Therefore, it is recommended to keep the values of hidden and read-only parameters untouched.

7.7. List of parameters by component

Device – Component setting video and audio inputs:

Sub section :	Parameter Name :	Description :
Video Input		
	Standard	Input format.
	Enable HSBC	Allow adjusting video characteristic when Enable HSBC is on. Do not enable this parameter if you want to keep the same colors as video source.
	Contrast	Contrast Gain value when Enable HSBC is on
	Brightness	Brightness Gain value when Enable HSBC is on
	Saturation	Saturation Gain value when Enable HSBC is on
	Hue	Hue Gain value when Enable HSBC is on
	Sharpness	Sharpness value when Enable HSBC is on
	Gamma	Gamma correction value when Enable HSBC is on
Sub section :	Parameter Name :	Description :
Audio Input		
	Source	SDI Embedded, AES or Analog input
	Analog Volume Left	Left volume gain for Analog Audio in dB.
	Analog Volume Right	Right volume gain for Analog Audio in dB.

	Digital Audio Gain	Audio gain for audio AES/EBU and audio embedded inputs in dB.
Sub section :	Parameter Name :	Description :
SDI Video Output		
	Output Mode	Framebuffer allows to get the same signal as encoded video (i.e no audio in output but OSD is present). SDI Passthrough means that video output signal is the same as input
	OSD	Turn on or off OSD display in SDI video output.
	Enable SDI pattern generator	Enable standard pattern on SDI output of the card
Sub section :	Parameter Name :	Description :
Composite Video Output		
	Output Mode	Framebuffer allows to get the same signal as encoded video (i.e OSD is present). Passthrough means that video output signal is the same as input.
	Standard	Set Standard format for the board output
	OSD	Turn on or off OSD display in composite video output.
	Enable WSS	Enable Wide Screen Signaling signal in composite output. Only operational when composite output is configured in PAL format.
	WSS Aspect Ratio	Select Aspect Ratio format information given embedded in Wide Screen Signaling format
	Enable Color bar	Enable standard pattern on composite output of the card
	Disable output	Disable composite output
	Keep Input Aspect Ratio	Keep 16/9 Aspect ratio in composite output from video source by adding black horizontal bars when enabled.
Sub section :	Parameter Name :	Description :
Preview Channel		
	Output Mode	Framebuffer allows to get the same signal as encoded video (i.e OSD is present). Passthrough means that video preview signal is the same as input. It avoids preview flickering when configuring the encoder
	OSD MASTER	Turn on or off OSD display in video preview.
	Downscaling FD1 On/Off	Downscale video preview signal to Standard Definition. Disabling this parameter improves resolution of video preview but it may result in missing frame in video preview or bad effects depending on system performances.
Sub section :	Parameter Name :	Description :
Video Capture Channel		
	OSD	Turn on or off OSD objects in all encoding channels.
Sub section :	Parameter Name :	Description :
Still Image Capture Channel		
	OSD	Turn on or off OSD objects in bmp files.

Sub section :	Parameter Name :	Description :
Capture Channel		
	Audio Encoder Channel 0	Select which embedded stereo audio channels will be recorded in first audio track of encoded file..
	Audio Encoder Channel 1	same for 2 nd track
	Audio Encoder Channel 2	same for 3 rd track
	Audio Encoder Channel 3	same for 4 th track
	Audio Encoder Channel 4	same for 5 th track
	Audio Encoder Channel 5	same for 6 th track
	Audio Encoder Channel 6	same for 7 th track
	Audio Encoder Channel 7	same for 8 th track
Sub section :	Parameter Name :	Description :
None		
	CC Extraction	Allow to extract close caption data from VBI signal and insert it in SEI message of AVC stream.
	TC Extraction	Allow to extract timecode data from VBI signal and insert it in Pict Timing SEI message of AVC stream.
	TC Encoded Event	Allow to retrieve inserted timecode in AVC stream in application through an event.
	TC Input Extraction Event	Allow to retrieve timecode captured from VBI signal in application through an event.
	Enable VBI ReOrdering	Allow to reorder timecode in Presentation Order.
	Vbi Encoding Offset	Set accuracy of timecode captured from VBI .
	SDI Camera	Read-only, contains detected format on SDI input (0 means no detected signal)
	Analog Camera	Read-only, contains detected format on composite input (0 means no detected signal)

HW Video Pre-Processor A (or B)– Component capturing video (do not change default parameters):

This component is hidden since all these parameters are hidden and should not be modified.

VM4400 AVC Encoder A (or B)– Component encoding video stream:

Parameter Name :	Description :
Format Video/Resolution	Select Video Format and framerate in frames per seconds (resolution @frames per second). For instance

	720p60 becomes 720p@60 1080i60 becomes 1080i@30
Aspect Ratio	Set Aspect Ratio (auto value use same value as source)
Profile	Select Encoder Profile
Level	Select Encoder Level
Bitrate	Set Average bitrate in VBR mode
Max Bitrate	Set maximum bitrate in VBR mode and average in CBR mode
GOP Structure	Set GOP structure
GOP Size	Set Distance between two I frame
Enable CABAC	Not operational
Initial Delay	Set latency of encoder in number of frame
Enable CBR	0 - Enable VBR mode 1 - Enable CBR mode
Low Delay Mode	Allow low delay mode.
Closed GOP	Force every I frame as IDR
SPS/PPS per IDR	Enable PPS info header for each IDR
Picture coding	Set Picture coding format
Picture Timing SEI Messages	Add Picture Timing SEI Messages in AVC stream.
Insert Timestamp in Picture Timing SEI Messages	Insert Timestamp in Picture Timing SEI Messages
Scene Change	Enable Scene change detection
QP base	Set first quantisation parameter
QP Range Min	Set smallest quantisation parameter. A higher value will reduce maximum size of frames but it Increases number of dropped frames.
QP Range Max	Set highest quantisation parameter
Fade Detect Enable	Allow encoder to change QP when a fade is detected.
QP Fade Delta	sets quantisation parameter delta variable applied during fades (when Fade Detect Enable is on)
Insert AU Delimiters	Insert AU Delimiter in AVC stream
Blu-ray Compliance	Change Slice ID as described in Blu-ray standard
Enable drop frame compensation	Compensate a drop frame by a empty AU.

HW Audio Encoder A(or B) – Component encoding audio stream:

Parameter Name :	Description :
Codec Type	Audio encoder mode
Audio Bitrate	Encoding bitrate
Audio Sample Rate	Input sample rate
Audio Channel Number	Audio mode
Audio Sample Size	Audio sample accuracy

Vitec Multiplexer A (or B) – Component multiplexing Video and Audio encoded streams:

Parameter Name :	Description :
Ts Packet Size	Set packet size for streaming (188) or file storage (192)
Max PAT Update Time	Set Max PAT Update Time
Max Number of Write	Set Max Number of Write

Video PID	Set Video PID
Audio PID	Set Audio PID
Private PID	Set Private PID
Extra PID	Set Extra PID
Program Map Or Network Pid	Set Program Map Pid
Enable CBR	Enable Constant Bitrate (output size = Video bitrate * (1+MuxRate OverHead Percent) * Duration in seconds. When off output size = Video bitrate * Duration
Initial PCR	Set Initial PCR
Initial Delay	Set Delay PCR
Maximum Delay	Set Maximum Delay
Minimum Delay	Set Minimum Delay
Max PCR Update Time	Set Max PCR Update Time
MuxRate OverHead Percent	Set MuxRate OverHead Percent when Enable CBR is 1
Enable Streaming	Enable Streaming or not

OSD MASTER – Component overlaying text or picture in video

Parameter Name :	Description :
Font	Set Font Text for overlaid text
Text	Set text to overlay
File Name	Set File Name of the image to overlay
OSD Mode	Must be on YUV
Output Format	Format of OSD (do not modify)
OSD Acceleration	Cache OSD caracters in memory
OSD Timer Granularity	OSD Timer Granularity

Still Image Capture – Component creating bmp files:

Parameter Name :	Description :
Copy To Clipboard	Store the still image into clipboard memory
Encoder Format	Set still image format (only BMP)
Deinterlace	Apply Vitec deinterlacing algorithm to bmp (should be used only in interlaced format)

Overlay monitor – Component overlaying an audio meter on top of video preview (right position):

Parameter Name :	Description :
Audio Meter On	Enable/Disable overlay monitor
Audio Meter Transparency	Transparency of overlay monitor
Rate Display On	Enable monitoring of output of Video Encoder. A white pixel is displayed for each frame. Its position depends on Frame size divided by Rate Display Scale

Rate Display Scale	Rate Display Scale
Audio level event On	Sent livewire events to application with audio levels in dB

Status monitor – Component overlaying the encoding elapsed time on top of video preview.

Parameter Name	Description
Status On	Enable/Disable status monitor
Count Pause	The elapsed time keeps increasing when encoding is paused
X	Horizontal position in the video preview
Y	Vertical position in the video preview
Special Effect	Visual effect on the status monitor
Special Effect Period	In ms, the period of the selected special effect
Text Transparency	Transparency of the status monitor
Background Transparency	Transparency of background of the status monitor
Text Color	Color of the value of the status monitor
Background Color	Color of the background of the status monitor

StreamRegulator A (or B)– Component that featuring pause/resume and auto split:

Parameter Name :	Description :
Cut each X seconds"	Set maximum duration of file. After X seconds value, a new file is created.

Storage A (or B) – Component writing encoding video file on hard disk:

Parameter Name :	Description :
Path	Set Output file name of encoding file.
Last Used Path	Get Output file name of last encoding file
Length	Get file size in Byte
Length in MB	Get file size in Mega Byte
Position	Get current offset in Byte

Still Image Storage A– Component writing bmp picture on hard disk:

Parameter Name :	Description :
Path	Set Output file name of encoding file.
Last Used Path	Get Output file name of last encoding file
Length	Get file size in Byte
Length in MB	Get file size in Mega Byte
Position	Get current offset in Byte

Video Preview – Component displaying video preview:

Parameter Name :	Description :
Mode	Display mode (DirectDraw or D3D) D3D can reduce tearing effect on windows 7 systems.
Frame Rate (1/N)	Show 1/N image
Show Both Fields	Disable Vitec deinterlacing algorithm on preview screen when value is off
Apect Ratio	Force Aspect Ratio of Video Preview
Deinterlace	Enable and Set Deinterlacing algorithm in preview. Deinterlacing should be disabled in video progressive format

7.8. Frequently Asked Questions:

7.8.1. What can I do if my board is not detected by the system?

The Vitec driver has to be listed in the Sound, Video and Game controller of the Device manager.

In Advanced details- Hardware ID, you should see for the Vendor ID "142E" value and for Device ID "7040" value.

If you can not find the Vitec driver, it means that the system has not detected the card.

Please try to move the Vitec card in a different slot.

7.8.2. Why the register.bat does not register properly the dlls file?

The registration is realized by regsvr32 windows utility.

To properly register the dlls, you must be administrator on the system.

You should also disable the User Account Control on last Windows versions.

On last versions of Windows, you must add some code at the beginning of register.bat in order to require Administrator in command line:

```
@echo off
```

```
:: BatchGotAdmin
```

```
:-----
```

```
REM --> Check for permissions
```

```
>nul 2>&1 "%SYSTEMROOT%\system32\cacls.exe"  
"%SYSTEMROOT%\system32\config\system"
```

```
REM --> If error flag set, we do not have admin.
```

```
if '%errorlevel%' NEQ '0' (
```

```
    echo Requesting administrative privileges...
```

```
    goto UACPrompt
```

```
) else ( goto gotAdmin )
```

```
:UACPrompt
```

```
    echo test
```

```
    echo Set UAC = CreateObject^("Shell.Application") > "%temp%\getadmin.vbs"
```

```
    echo UAC.ShellExecute "%~s0", "", "", "runas", 1 >> "%temp%\getadmin.vbs"
```

```
"%temp%\getadmin.vbs"
```

```

exit /B
:gotAdmin
    echo admin
    if exist "%temp%\getadmin.vbs" ( del "%temp%\getadmin.vbs" )
    pushd "%CD%"
    CD /D "%~dp0"

REM PASTE THE CONTENT OF REGISTER.BAT HERE

:-----

```

7.8.3. Why there is no preview when it is enabled?

Video preview is based on DirectDraw or Direct3D capabilities of the graphic card of the system. Some graphic card do not implement these features so video preview can not be displayed.

Please verify if a driver update of the graphic card fixes this technical issue or try to change the graphic card.

If it does not help, please verify that your video input settings are configured in the same format as your video source.

7.8.4. Why livewire application status becomes red when I start the encoding?

VMC-7440 can not encode if no source has been detected since the start of application. It is recommend filtering start command to Capture branch when no source is detected since it will result in a frozen image in case of removed video source or a critical error.

7.8.5. Why livewire application status becomes red when I start the encoding?

VMC-7440 can not encode if no source has been detected since the start of application. It is recommend filtering start command to Capture branch when no source is detected since it will result in a frozen image in case of removed video source or a critical error.

7.8.6. What to do if I get a blue screen when starting the livewire application?

This kind of issue is related to implementation of features in the BIOS of motherboard that can lead to a incompatibility with the card.

Sometimes, this kind of problem can be fixed by a BIOS update.

If this does not work, there are few workarounds to try:

- Change PCI Express Settings in the BIOS if this section exists, one by one and check the effect at boot for each modification.
- Disable PAE (see pae option of BCDEdit).

- Use the forcedisable option to override the advanced PCI Express features of Windows 7 (see pciexpress option of BCDEdit).
- Force legacy APIC mode (see uselegacyapicmode option of BCDEdit)
- Decrease memory size if equal to 4GB if operating system is 32 bits.

7.8.7. Why livewire sample application does not appear after starting it?

Application saves in the windows registry its last position on Desktop and last used XML file.

If application has been closed when minimized or outside the desktop or XML is corrupted, it will not be able to start.

In this case, kill the application through Windows' Task Manager, remove all Keys from "HKEY_CurrentUser/Software/Vitec Multimedia/livewire sample application" and restart it.

If the same problem occurs, please try to restore the original xml file from the SDK and register manually the dlls with register.bat.

7.8.8. Why livewire settings page does not appear after clicking on the Settings... button?

Please verify if SettingsPage.dll is present in the application directory. Please copy it from SDK if missing.

7.8.9. Why I need to reconfigure my card every time I start my application?

Please verify that you have write access on the used XML file.

7.8.10. Why do I have no video or bmp files when clicking on Start/Capture button?

Please verify that file names of MPEG or bmp files are located in an existing directory. Please also check that there is enough free space on hard disk.

7.8.11. Why my XML files has been corrupted?

Please note that you have to put the XML file in a different partition that partition where you record MPEG data.

Indeed, application saves all parameters modification done during the session. If the partition is full, this process will corrupt the XML file.

7.8.12. Why do I have time to time a bad video frame on my video preview and encoding?

Please note that capturing real time video on a PCI express card requires having good PCI transfer performances between the system and the Vitec card. Some low cost or badly designed motherboards can not reach this level of performance and produces time to time a bad video frame.

If you encounter this kind of behaviour on a system, please try to update the BIOS of the motherboard, move the card to a different slot or move the card to a different motherboard.

7.8.13. Why the beginning of the recoding is corrupted after switching a profile?

This is a known issue when switching from composite to SDI input preset. This issue is related to the transition from interlaced to progressive formats. First frames of the file show a damaged portion of the last captured frames. A possible workaround is to start a first encoding of 100 ms with a non existing filename in order to reset internal video buffer after a switch of profile. This encoding will be transparent for the user since no file will be created in this case.

7.8.14. What does represent the white dots in the video preview when I start encoding?

These white dots represent frame size and are generated by the Overlay Monitor component. It is possible to disable this option by turning off "Rate Display On" parameter of Overlay monitor section from the Livewire Settings Page.

7.8.15. What does represent the two green bars on the right side of video preview?

The two greens bars represent audio levels in dBFS captured from the audio source. Top of the audio level means 0 dBFS and bottom of the audio level is -98 dBFS. This information is given by Status Monitor component and can be retrieved through an audio event.

It is possible to disable this option by turning off "Audio Meter On" parameter of Overlay monitor component using Livewire Settings Page.

7.8.16. What can I do when there are some visible default on my video preview

Direct3D mode is the best mode for the display since it adds some functionality to improve the video quality and avoid tearing effect. Please try to switch the mode of the video preview into D3D.

7.8.17. Can I get events, warnings and errors from the product?

Livewire sample application does not display warning, error or event.

However, it is possible to receive all kind of event, notification, warning or error in a custom application by subscribing the application to Eventlog notification (see source of the livewire sample application).

Implementing Eventlog notification is very useful in case of troubleshooting. If your application does not subscribe to the Eventlog, you can ask for LogsFlowServer.exe tool that can trace all events from Livewire.

7.8.18. Why do I have some parameters in the XML file that application can not read or modify?

Some components have options which are not available for every product. It is possible to find in the XML file some parameters that are not implemented in your product version.

7.8.19. What are the requirements to set up my development environment?

VMC-7440 SDK supports Microsoft Visual Studio v6.0, VC2008 SP1 and VC 2010. VMC-7440 SDK can only produce 32 bits code that can run on 32 or 64 bits Microsoft operating system. Microsoft Visual C++ 2008 SP1 Redistributable Package and .NET framework 3.5 should be installed on the development system. The SDK is divided into two sub folders:

LiveWireSDK: This directory is required for building application based on Livewire technology.

The batch file "set_sdk.bat" sets required environment variables that refer to this directory.

LIVEWIRESDK_MSVC6 for Microsoft Visual 6

LIVEWIRESDK_MSVC9 for Microsoft Visual 9 and 10

LiveWireBaseDir : This directory is required for building custom Livewire components. It includes some binaries, libraries, interfaces and headers files.

The batch file "set_basedir.bat" sets two required environment variables that refer to this directory.

LIVEWIREBASEDIR_MSVC6 for Microsoft Visual 6

LIVEWIREBASEDIR_MSVC9 for Microsoft Visual 9 and 10

Please note that application project especially CSharp may refer to dlls present in \Product_Files\Bin folder.

7.8.20. How can I create my first custom component?

For developers that intent to implement their own algorithm in a dedicated Livewire component, a wizard for Microsoft Studio VC9/10 creating the skeleton of Livewire component that passes video data from its input to the output is available in the DevResources directory.

To install it, you have to copy content of \DevResources\ComponentWizardVC9 and run InstallWizard_ VC9.cmd. You will be able to create a custom component for Microsoft Visual Studio 2010.

Each custom component must be manually added to the component section and assembly section of XML configuration file in order to be used. It also has to be

registered in Windows' Registry. Please refer to LiveWire Implementing Custom Component.html document from \DevResources\Tutorial for further information.

7.8.21. How can I change a parameter from my application?

IAssemblyContainer interface has some methods that allow interacting directly with the parameter of component.

```
get_ComponentParameter(BSTR bstrComponentName, BSTR bstrParameterName, LONG* pIVal);
```

This method returns current integer value parameter in pIVal variable.

```
put_ComponentParameter(BSTR bstrComponentName, BSTR bstrParameterName, LONG IVal);
```

This method sets current integer value to IVal.

```
get_ComponentParameterText(BSTR bstrComponentName, BSTR bstrParameterName, BSTR* pbstrVal);
```

This method returns current text value parameter in pbstrVal variable.

```
put_ComponentParameterText(BSTR bstrComponentName, BSTR bstrParameterName, BSTR bstrVal);
```

This method sets current text value bstrVal argument.

Please note that for parameters which are in a sub section (see paragraph 5.4), main component and subsection must be specified following path rule:

`bstrComponentName = L"MainComponent\\Subsection".`

For instance to modify Contrast parameters of Device it is necessary to add Video Input section with this call.

```
m_pAssemblyContainer->put_ComponentParameter(L"Device\\Video Input", L"Contrast", new_value);
```

7.8.22. Where can I find access rights, current, default, min and max value for each parameter?

Parameter values are stored in the XML file in profile sections with their values and a flag for access rights management.

Each bit of the AccessFlag represents different rights that can be combined.

1 = Read-only mode.

2 = Hidden mode (only visible from the Livewire Setting page when Show Hidden is checked in the view menu).

8 = Protected mode (similar to read only except that right is cleared when creating a child profile from the preset).

Other bits are reserved.

For integer parameters, there is current, default, max and min values described as follow

```
<Property AccessFlags="0" Current="XXX" Default="XXX" Max="XXX" Min="XXX" Name="Parameter_name"/>
```


7.8.23. Why a circle display on TV is not a circle in BMP file?

It is important to realize that NTSC and PAL monitors have rectangular pixels whereas computer monitors typically have square pixels.

For NTSC, the VMC-7440 card captures 720x480 pixels by image. This image displays in a 720x480 windows resolution is larger than expected so it is necessary to display it in a 640x480 window to be closer than a display on a TV.

For PAL, the VMC-7440 card captures 720x576 pixels by image. This image displays in a 720x576 windows resolution is narrower than expected so it is necessary to display it in a 768x576 to be closer than normal a display on a TV.

When creating picture for OSD, please note that you need to load a 720x576 picture for PAL and 720x480 for NTSC whatever the size of the video preview.

7.8.24. Why my tga picture is displayed upside down after loading it into the OSD buffer on Video Preview?

The VMC-7440 card support only one uncompress Truevision format which expects having origin of the picture in lower left-hand corner.

It means that TGA file must have the bit 5 at offset 0x17 equals 0.

7.8.25. How can I use several VMC-7440 cards in the same system?

It is possible to control up to 16 VMC-7440 cards in the same system.

Three different solutions are possible for controlling several cards.

- First solution: load one instance of application per board.

This is the quickest solution to set up. It requires 1 XML file per board and one instance of application.

In this case, developer has to duplicate the standard xml file, edit it with a text editor and replace the CLSID value of the VMC-7440 component in the component section with a text editor.

The line to modify is the following one:

```
<Component CLSID="1F3E09CB-96AC-43BC-87F0-AC9F24F5B0EE" Name="VMC-7440">
```

You have to replace 1F3E09CB-96AC-43BC-87F0-AC9F24F5B0EE by:

1F3E09CB-96AC-43BC-87F0-AC9F24F5B011 - for the 2nd card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B012 - for the 3rd card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B013 - for the 4th card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B014 - for the 5th card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B015 - for the 6th card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B016 - for the 7th card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B017 - for the 8th card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B018 - for the 9th card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B019 - for the 10th card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B01A - for the 11th card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B01B - for the 12th card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B01C - for the 13th card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B01D - for the 14th card

1F3E09CB-96AC-43BC-87F0-AC9F24F5B01E - for the 15th card
 1F3E09CB-96AC-43BC-87F0-AC9F24F5B01F - for the 16th card

When it is done, user can start one instance of the livewire sample application per card from a terminal console and gives the xml path as first argument.
 For instance, if you would like to control the second board, you will have to run "livewire sample application.exe VM4-2-C7_osd_2nd.xml" from a terminal.
 Please avoid starting several livewire sample application on the same card in order to avoid any hardware conflict which may cause system crash.

- Second solution:

Develop application which create one instance of Assembly Container per board and initialize each with proper instance XML from the first solution.

- Third solution:

1 shared instance of application for all the board.

This solution requires creating a multiboarding XML file where every components, branches and parameters profiles are duplicated. Moreover, it requires developing an application different from the livewire sample application. To control different boards/branches for multi instance configuration

Please also verify that system has enough memory and CPU resources to support several instances.

7.8.26. Why does VLC media player freeze after decoding few seconds of playback?

VLC is sensitive to wrong PTS/DTS values.

It appears that PTS/DTS timestamps values are bad when Muxrate Overhead (%) parameter is not set properly in constant bitrate.

Please verify that multiplexer is in variable rate control. If you have to use constant rate control for the TS multiplexer please use the recommended value of 50%.

7.8.27. Does this SDK works with previous version of SDK?

Livewire have changed recently from version 3 to version 4 in order to bring new features such as possibility to change language of settings and Segment profile.

Please find below the list of changes.

Livewire V3	Livewire V4
<i>Format of Notify function of IBaseSubscriber interface changed</i>	
// Notification CallBack Class type CustomSubscriber = class(TInterfacedObject, IBaseSubscriber)	// Notification CallBack Class type CustomSubscriber = class(TInterfacedObject, IBaseSubscriber)

public procedure Notify(NCode: Integer); safecall; end;	public function Notify(NCode: Integer):HRESULT; stdcall; end;
<i>IAssemblyContainer interface replaced with IAssemblyContainerV4</i>	
IAssemblyContainer	IAssemblyContainerV4
IAssemblyContainer StorageFileName access changed from property to methods	
m_pAssemblyContainer.StorageFileName:= 'ActiveProfile.xml'; BrwDialog.FileName:=m_pAssemblyContainer.St orageFileName;	m_pAssemblyContainer.put_StorageFileName('Activ eProfile.xml'); BrwDialog.FileName:=m_pAssemblyContainer.get_S torageFileName();
IAssemblyContainer Profile access changed from property to methods	
m_pAssemblyContainer.ProfileName:='MPEG1 low';	m_pAssemblyContainer.put_Profile('Encoder', 'MPEG1 low');
Branch state enum names updated	
BST_RUN BST_STOP ...	BST_STATE_RUN BST_STATE_STOP ...
Parameter access interface upgraded from X_Parameter to X_PARAMETER_V4	
pxParameter: X_Parameter;	pxParameter: X_PARAMETER_V4; Note: X_PARAMETER_V4 interface behavior is not operational when application is written in Delphi. For instance, get_Name() function returns an empty string.
Components parameters access is unified.	
<p>Custom parameters interfaces are discarded.</p> <p>Parameters are accessible via two pairs of functions of IAssemblyContainer interface</p> <p>IDL description of parameters access functions.</p> <p>// numeric parameter access</p> <pre> HRESULT get_ComponentParameter([in] BSTR bstrComponentName, [in] BSTR bstrParameterName, [out, retval] LONG* pIVal); HRESULT put_ComponentParameter([in] BSTR bstrComponentName, [in] BSTR bstrParameterName, [in] LONG IVal); </pre> <p>// text parameter access</p>	

```

HRESULT get_ComponentParameterText([in] BSTR
bstrComponentName, [in] BSTR
bstrParameterName, [out, retval] BSTR* pbstrVal);

HRESULT put_ComponentParameterText([in] BSTR
bstrComponentName, [in] BSTR
bstrParameterName, [in] BSTR bstrVal);

```

7.8.28. How can I monitor encoded duration or number of encoded frames with VMC-7440?

SDK from version 8.0.5 implements IAuxTime interface that returns the encoding duration in ms or number of frames through 2 methods:

- get_FrameNumber(\$IValueFrame).
- get_Milliseconds(\$IValueMs).

IAuxTime interface can be used on “**AVC HW Encoder A**” for the first video channel and on “**AVC HW Encoder B**” for the second video channel.

We delivered in version 8.0.6 a specific version of livewire sample application that calls these 2 methods to “AVC HW Encoder A” component instead of “Video Encoder” component in the Notify function.

(see line 1733 of livewire test applicationdIlg.cpp).

```

if (m_pAssemblyContainer) {
    IComponentContainerV4* pComponentContainer = NULL;
    HRESULT hr = m_pAssemblyContainer-
>QueryInterface(IID_IComponentContainerV4, (void**)&pComponentContainer);
    if (SUCCEEDED(hr)) {
        IAuxTime* piAT = NULL;
        if (SUCCEEDED(hr = pComponentContainer-
>QueryComponentInterfaceByName(L"AVC HW Encoder A", IID_IAuxTime,
(IUnknown**)&piAT))) {
            LONGLONG IValueFrame = 0;
            LONGLONG IValueMs = 0;
            LONGLONG IValueTc = 0;
            piAT->get_FrameNumber(&IValueFrame);
            piAT->get_Milliseconds(&IValueMs);
            piAT->get_TimeCode(&IValueTc);
            piAT->Release();
            TCHAR str[256];
            wsprintf(str, "Segment length = %d frames, %d ms, %08X\n",
(LONG)IValueFrame, (LONG)IValueMs, (LONG)IValueTc);
            OutputDebugString(str);

```

```

    }

    pComponentContainer->Release();

}

}

```

To see how it works, you have just to set a capture duration (like 30s) in the graphic interface, open DebugView utility to watch the OutputDebugString, start a Capture and wait for the following messages:

Segment Length = XXX frames, XXXXX ms, 00000000

7.8.29. Can I change the language of the Livewire Settings page?

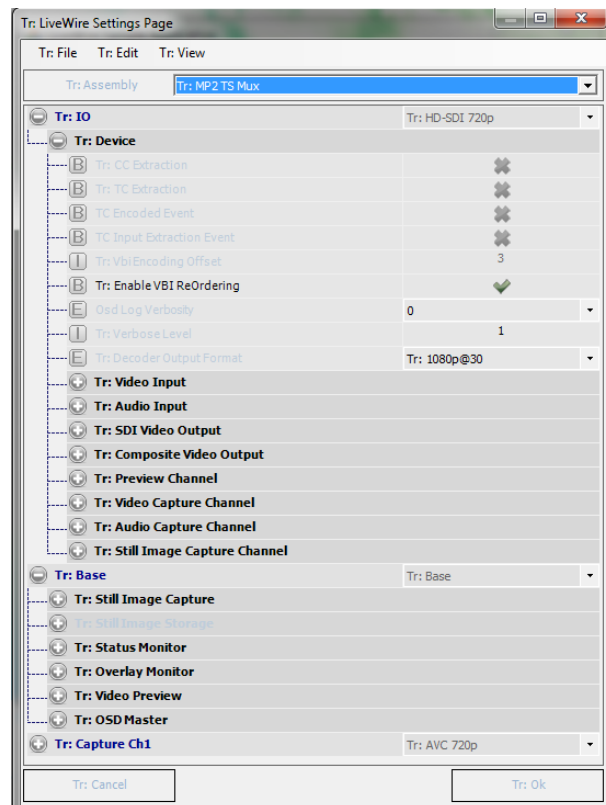
The livewire settings page supports localization. It means that it is possible to customize the display name of every text displayed to support a new language.

For each new language, it is necessary to create a XML file containing the dictionary that translates the parameter name in the new language.

All dictionary XML has to be in Unicode standard that offers the possibility to display characters of any alphabet.

By default, a VMC7440 dictionary EN_example.xml is provided.

It takes every text name item and adds "Tr: "prefix of every of them. (see below)



The dictionary XML file is structured as bellows:

- **LocalizationSection**

- **ComponentsSection**

Description:

This sub section contains for every component name the display text of every parameters names and text values.

Syntax:

```
<Component Name="Component Name">
<Property Name="Parameter Name" DisplayText="Displayed Parameter Name">
<Value Index="Value Name" DisplayText="Value Name " />
</Property>
</Component>
```

- **AssemblySection**

Description:

This sub section contains for every assembly the display text of every Segment Profiles and component.

- **Assembly Name**

Syntax: <Assembly Name="Assembly Name">
</Assembly>

- **SegmentList**

Syntax:

```
<Segment Name=" Segment Name " DisplayText=" Displayed
Segment Name ">
<Profile Name=" Profile Name» DisplayText="Displayed Profile Name "
/>
</Segment>
```

- **MemberList**

Syntax:

```
<Member Name="Device" DisplayText="Tr: Device" />
```

- **CustomDictionarySection**

Description: This section contains for each text item of user interface, assembly, Segment or Profile, the custom display text.

Syntax:

```
<CustomDictionarySection>
</CustomDictionarySection>
```

- **UI**

Syntax:

```
<Section Name="UI">
<Value Name="mnu_SettingsForm_File" DisplayText="Tr: File" />
```

```

<Value Name="mnu_SettingsForm_ActionRecorder" DisplayText="Tr:
Journaling" />
<Value Name="mnu_SettingsForm_Cancel" DisplayText="Tr: Undo All and
Exit" />
<Value Name="mnu_SettingsForm_Ok" DisplayText="Tr: Exit" />
<Value Name="mnu_SettingsForm_Edit" DisplayText="Tr: Edit" />
<Value Name="mnu_SettingsForm_ClearUndo" DisplayText="Tr: Clear
Undo Queue" />
<Value Name="mnu_SettingsForm_Undo" DisplayText="Tr: Undo" />
<Value Name="mnu_SettingsForm_View" DisplayText="Tr: View" />
<Value Name="mnu_SettingsForm_ShowHidden" DisplayText="Tr: Show
Hidden" />
<Value Name="mnu_SettingsForm_HexView" DisplayText="Tr: Hex View" />
<Value Name="mnu_SettingsForm_ShowHints" DisplayText="Tr: Show
Hints" />
<Value Name="mnu_SettingsForm_FontSize" DisplayText="Tr: Font Size" />
<Value Name="mnu_SettingsForm_FontSize_Small" DisplayText="Tr:
Small" />
<Value Name="mnu_SettingsForm_FontSize_Medium" DisplayText="Tr:
Medium" />
<Value Name="mnu_SettingsForm_FontSize_Large" DisplayText="Tr:
Large" />
<Value Name="lbl_SettingsForm_Assembly" DisplayText="Tr: Assembly " />
<Value Name="btn_SettingsForm_Cancel" DisplayText="Tr: Cancel" />
<Value Name="btn_SettingsForm_Ok" DisplayText="Tr: Ok" />
<Value Name="mnu_SettingsForm_Hidden" DisplayText="Tr: Hidden" />
<Value Name="mnu_SettingsForm_ReadOnly" DisplayText="Tr: Read Only"
/>
<Value Name="mnu_SettingsForm_SetDefault" DisplayText="Tr: Set
Default" />
<Value Name="SettingsForm" DisplayText="Tr: LiveWire Settings Page" / --
>
</Section>

```

- **Assembly**

Syntax:

```

<Section Name="Assembly">
<Value Name=" Value Name DisplayText="Display Value Name " />
</Section>

```

- **Segment**

Syntax:

```
<Section Name="Segment">
<Value Name="IO" DisplayText="Tr: IO" />
</Section>
```

- **Profile**

Syntax:

```
<Section Name="Profile">
<Value Name=" Value Name " DisplayText=" Display Value Name" />
</Section>
```

Each dictionary XML file has to be declared in ConfigurationSection section of the VMC-7440 reference.xml file as below:

```
<ConfigurationSection>
<Property AccessFlags="0" Current="0" Default="0" Name="Dictionary
XML">
<Value Index="0" Text="VMC7440 dictionary EN.xml" />
<Value Index="1" Text="VMC7440 dictionary FR.xml" />
<Value Index="2" Text="VMC7440 dictionary RU.xml" />
<Value Index="3" Text="VMC7440 dictionary CH.xml" />
</Property>
</ConfigurationSection>
```

The current dictionary can be change through an interface of IAssembly Container.